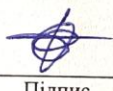



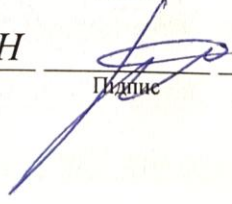
## КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

на тему Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

Галузь знань 12 – Інформаційні технології  
Шифр і назва галузі знань  
Спеціальність 122 – Комп'ютерні науки  
Шифр і назва спеціальності  
Освітня програма Комп'ютерні науки  
Назва освітньої програми

Виконав: студент групи КН-21-2  Назар ОСТАПЧЕНКО  
Група виконавця Підпис Ім'я, ПРІЗВИЩЕ

Керівник: асистент каф. КН  Олена ТИЩЕНКО  
Науковий ступінь, посада Підпис Ім'я, ПРІЗВИЩЕ

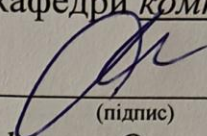
Нормоконтроль: к.т.н., доц. каф. КН  Руслан БАГРІЙ  
Науковий ступінь, посада Підпис Ім'я, ПРІЗВИЩЕ

До захисту допускаю:  
Зав. кафедри КН, д.т.н., професор  Олександр БАРМАК  
Підпис Ім'я, ПРІЗВИЩЕ

19 06 2025 р.

ХМЕЛЬНИЦЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
Факультет інформаційних технологій  
Кафедра комп'ютерних наук  
Освітній ступінь бакалавр  
Галузь знань 12 – Інформаційні технології  
Спеціальність 122 – Комп'ютерні науки

ЗАТВЕРДЖУЮ  
Завідувач кафедри комп'ютерних наук

  
\_\_\_\_\_ (підпис)

д.т.н., професор Олександр БАРМАК  
« 10 » 02 2025 року

### ЗАВДАННЯ НА КВАЛІФІКАЦІЙНУ РОБОТУ БАКАЛАВРА

1. Тема кваліфікаційної роботи бакалавра: «Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних»

2. Завдання видано студенту \_\_\_\_\_ Назару Остапченко  
(Ім'я, прізвище)

3. Керівник роботи \_\_\_\_\_ асистент кафедри КН Олена Тищенко  
(посада, ім'я, прізвище)

4. Затверджено наказом університету від « 07 » \_\_\_\_\_ 02 2025 р. № 23

5. Дата видачі завдання студенту: « 10 » \_\_\_\_\_ 02 2025 р.

6. Зміст пояснювальної записки (перелік задач) та вихідні дані:

Мета роботи – покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних. Для досягнення мети слід виконати такі задачі: виконати дослідження предметної області для задачі пошуку релевантних зображень; розробити метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних; здійснити програмну реалізацію інтелектуальної системи автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних та провести дослідження роботи методу з використанням розробленої інтелектуальної системи.

7. Календарний план виконання кваліфікаційної роботи бакалавра:

№	Назва етапів (розділів) кваліфікаційної роботи бакалавра	Термін виконання	Примітка
1	Вибір напряму дослідження та узгодження тематики кваліфікаційної роботи бакалавра з керівником, складання календарного графіка виконання	січень 2025	Виконано
2	Ознайомлення з предметною областю, формулювання мети і задач дослідження, визначення об'єкта та предмета дослідження	лютий 2025	Виконано
3	Проектування та розроблення методу вирішення завдання, загальної архітектури програмного забезпечення, інтерфейсу користувача, вибір засобів реалізації програмного забезпечення	березень 2025	Виконано
4	Створення та тестування програмного забезпечення, дослідження ефективності, висновки з виконаної роботи	квітень 2025	Виконано
5	Написання пояснювальної записки, урахування зауважень керівника, оформлення згідно з вимогами	травень 2025	Виконано
6	Розробка презентаційних матеріалів та попередній захист кваліфікаційної роботи	травень 2025	Виконано
7	Отримання відгуку керівника, рецензії, перевірка на плагіат, нормоконтроль	червень 2025	Виконано
8	Підготовка до захисту та захист кваліфікаційної роботи	червень 2025	Виконано

Виконавець: студент групи КН-21-2

Група виконавця

  
Підпис

Назар ОСТАПЧЕНКО

Ім'я, ПРІЗВИЩЕ

Керівник:

асистент каф. КН

Науковий ступінь, посада

  
Підпис

Олена ТИЩЕНКО

Ім'я, ПРІЗВИЩЕ

## Анотація

Тема кваліфікаційної роботи бакалавра: «Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних»

Виконавець кваліфікаційної роботи бакалавра: студент групи КН-21-2 Назар Остапченко

Керівник кваліфікаційної роботи бакалавра: асистент каф. КН Олена Тищенко

Кваліфікаційна робота бакалавра містить:

Пояснювальна записка				Кількість додатків
Сторінок	Рисунків	Таблиць	Джерел інформації	
57	22	2	46	6

Метою кваліфікаційної роботи бакалавра є покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних. Для розробки інтелектуальної системи було використано мову програмування Python, а також спеціалізований CLIP, для порівняння векторів та пошуку релевантних зображень для навчання та використання класифікатора. Розроблена інтелектуальна система призначена для використання в комерції, для пошуку подібних товарів, та пошуку об'єктів, коли користувач не знає точної назви.

Напрямами практичного використання розробленої інтелектуальної системи визначено пошук релевантних зображень.

Ключові слова: семантичний пошук, зображення, вектор, CLIP, ChromaDB.

Виконавець: студент групи КН-21-2

Група виконавця



Підпис

Назар ОСТАПЧЕНКО

Ім'я, ПРІЗВИЩЕ

## Зміст

Перелік скорочень .....	3
Вступ.....	4
Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій.....	6
1.1 Аналіз інформаційних моделей в області автоматизованого підбору зображень за запитом.....	6
1.2 Засоби інтелектуального підбору релевантних зображень .....	8
1.3 Аналіз наукових праць та існуючих програмних засобів для підбору релевантних зображень .....	12
1.4 Мета та задачі кваліфікаційної роботи бакалавра .....	17
Розділ 2 Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних .....	18
2.1 Спосіб отримання нейромережевої моделі для реалізації семантичного пошуку .....	18
2.2 Схема та кроки методу автоматизованого підбору релевантних зображень засобами векторних баз даних .....	19
2.3 Проєктна архітектура та взаємозв'язок компонентів інтелектуальної системи автоматизованого підбору релевантних зображень .....	21
2.4 Схема та кроки процесів векторизації та анотації даних.....	25
2.5 Датасет для донавчання нейромережевої моделі .....	28
2.6 Статистичні метрики .....	30
2.7 Висновки до розділу 2 .....	33
Розділ 3 Експериментальне дослідження .....	35
3.1 Засоби реалізації інтелектуальної системи автоматизованого підбору релевантних зображень .....	35
3.2 Особливості використання спеціалізованих програмних компонентів .....	37
3.3 Структура інтелектуальної системи.....	39
3.4 Особливості реалізації інтелектуальної системи.....	41

	2
3.5 Експериментальне дослідження запропонованого методу .....	45
3.6 Висновки до розділу 3 .....	49
Загальні висновки.....	51
Перелік посилань.....	53
Додатки	

## Перелік скорочень

Скорочення, термін, позначення	Пояснення
БД	База даних
CBIR	Пошук за вмістом зображення
ІТ	Інформаційні технології
КН	Комп'ютерні науки
CNN	Згортова нейронна мережа
ПЗ	Пояснювальна записка
ПП	Програмний продукт
RAG	Retrieval-Augmented Generation
LLM	Велика Мовна Модель
wnid	WordNet ID
RGB	Red Green Blue
CLIP	Contrastive Language–Image Pretraining
BLIP	Bootstrapping Language-Image Pre-training
KNN	K-Nearest Neighbors
API	Application Programming Interface

## Вступ

Кваліфікаційна робота бакалавра присвячена покращенню процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних.

**Актуальність.** Пошук зображень є фундаментальним елементом сучасного світу, та важливою складовою інтернету, що забезпечує швидкий та інтуїтивно зрозумілий доступ до великого простору візуального вмісту. Завдяки прогресу штучного інтелекту та технологій машинного навчання пошук зображень продовжує розвиватися, стаючи дедалі складнішим і здатним задовольняти потреби користувачів усе точнішими та надійнішими способами.

Ще не так давно, найпростішим способом виконати пошук релевантного зображення було використання звичайної пошукової системи та введення текстового запиту, та додавання слова «зображення», але завдяки загальній еволюції технології зараз існують ефективніші способи пошуку зображень, тобто використання відповідних систем пошуку зображень або використання функції зворотного пошуку.

Зараз користувач може скористатись пошуком за зображенням, такого сервісу, як Google, і отримати результати пошуку, які відобразатимуть пов'язані зображення, продукти, веб-сайти, що використовують подібні зображення, а також результати з вмістом, пов'язаним із зображенням за допомогою технології класифікації зображень. Функції візуального пошуку та API візуального пошуку додаються всюди, де є функція пошуку: Google, Pinterest, Instagram, Amazon і Microsoft впровадили функції візуального пошуку. Візуальний пошук уже має застосування в електронній комерції, соціальних мережах, веб-сайтах, програмах і навіть у реальному житті.

Планування та оптимізація для візуального пошуку сьогодні важливіші, ніж будь-коли. Це не замінить текстовий пошук, але комбінація текстового, голосового та візуального пошуку служить одній меті – зробити життя користувача легшим і зручнішим. Використання зображень у пошуку

продовжуватиме зростати, і, що важливіше, продовжуватимуть з'являтися нові технології для візуального пошуку.

**Об'єкт дослідження** – процес автоматизованого підбору релевантних зображень на основі семантичного пошуку.

**Предмет дослідження** – методи та технології семантичного пошуку для автоматизованого підбору релевантних зображень.

**Мета кваліфікаційної роботи бакалавра** – покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних.

**Завдання кваліфікаційної роботи бакалавра** – виконати дослідження предметної області для задачі пошуку релевантних зображень; розробити метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних; виконати програмну реалізацію створеного методу автоматизованого підбору релевантних зображень на основі семантичного пошуку даних та провести дослідження роботи методу з використанням розробленої інтелектуальної системи.

## Розділ 1 Характеристика предметної області: аналіз моделей, методів та реалізацій

### 1.1 Аналіз інформаційних моделей в області автоматизованого підбору зображень за запитом

В повсякденному житті, людина використовує свій зір, для візуального пошуку, як основний спосіб орієнтації у просторі та взаємодії з навколишнім середовищем.

В комп'ютерних технологіях, візуальний пошук це технологія, як дозволяє здійснювати пошук інформації на основі аналізу зображень замість традиційних текстових запитів. Вона базується на методах комп'ютерного зору та штучного інтелекту, що забезпечують ідентифікацію та класифікацію об'єктів, наявних у цифрових зображеннях [1].

Використання великих мовних моделей (Large Language Models) та мультимодальних систем дозволяє інтегрувати обробку текстової та візуальної інформації для підвищення точності результатів. Технологічна основа візуального пошуку полягає у застосуванні алгоритмів глибокого навчання, зокрема згорткових нейронних мереж (CNN), які здатні виділяти характерні ознаки об'єктів на зображеннях.

Виділення ознак – це процес перетворення сирих, часто неструктурованих даних у змістовні ознаки, для подальшого навчання моделей машинного навчання (рисунок 1.1). Точність і ефективність цих моделей залежить від якості вхідних ознак [2].

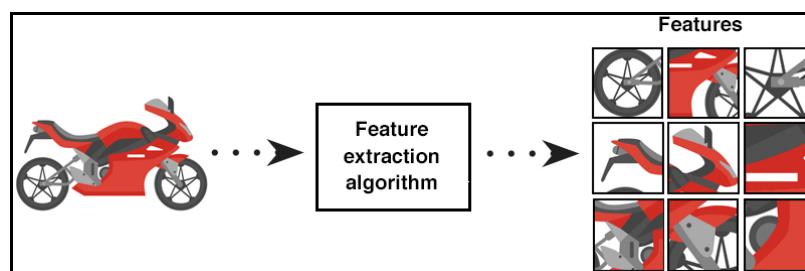


Рисунок 1.1 – Приклад виділення ознак об'єкта [3]

Наприклад, на зображенні вище подано необроблене вхідне зображення мотоцикла до алгоритму вилучення ознак. Цей алгоритм створює вектор, який містить перелік ознак і є одновимірним масивом, що характеризує об'єкт.

Варто зазначити, що на зображенні вище показано ознаки, вилучені лише з одного мотоцикла. Важливою властивістю ознак є повторюваність. Тобто ознака має дозволяти розпізнавати мотоцикли загалом, а не лише конкретний екземпляр чи модель. Коли алгоритм вилучення ознак бачить тисячі зображень мотоциклів, він розпізнає шаблони, які визначають, як приклад – колеса, в загальному вигляді, незалежно від їхнього розташування на зображенні чи типу мотоцикла.

Спеціалізовані пошукові системи, зокрема в області пошуку зображень, є одними з найшвидше зростаючих сервісів в Інтернеті. Подібно до текстового пошуку, ці системи є інформаційно-пошуковими системами, що допомагають користувачу знаходити інформацію в інтернеті за допомогою ключових слів або пошукових фраз [4].

Втім, більшість систем не використовують вищезазначене виділення ознак та змісту зображення, а індексують метадані (опис, теги, атрибути), збережені в базі даних. Під час виконання запиту пошукова система звертається до індексу, порівнює запит з наявними метаданими і видає результати за релевантністю.

Деякі системи здатні автоматично ідентифікувати обмежений набір візуального контенту, наприклад, обличчя, дерева, небо, будівлі, кольори, що може використовуватися як самостійно (content-based image retrieval), так і для доповнення метаданих.

Користувачу зазвичай надається набір мініатюр зображень, відсортованих за релевантністю, кожна з яких є посиланням на оригінальне джерело. Додатково користувач може використовувати розширені параметри пошуку для уточнення критеріїв: тип зображень (статичні, анімації), колірна гама, розмір зображення тощо.

Отже, актуальним залишається покращення процесу автоматизованого підбору релевантних зображень, яке є перспективним напрямом, що продовжує розвиватися завдяки удосконаленню алгоритмів машинного навчання та збільшенню обсягів доступних даних.

## **1.2 Засоби інтелектуального підбору релевантних зображень**

Існує ряд засобів інтелектуального підбору релевантних зображень, серед яких: семантичний пошук, пошук за вмістом зображення (CBIR), гібридні методи, які поєднують текст і візуальні дані, класифікаційні системи на базі машинного навчання, а також генеративні моделі, що створюють зображення у відповідь на текстовий запит.

Раніше традиційні пошукові системи зазвичай були зосереджені на зіставленні ключових слів у пошуковому запиті з відповідними ключовими словами на проіндексованих веб-сторінках, без глибшого розуміння контексту. Втім, існує метод, який дозволяє використати зміст запиту, для ефективнішого пошуку – семантичний пошук.

Семантичний пошук – це метод пошуку даних, який зосереджується на розумінні контекстуального значення та наміру пошукового запиту користувача, а не лише на відповідності ключових слів. Замість того, щоб просто шукати збіги між пошуковими запитами та проіндексованим вмістом, він націлений на надання більш релевантних результатів пошуку, враховуючи різні фактори, зокрема зв'язки між словами, місцезнаходження користувача, будь-які попередні пошуки та контекст пошуку [5].

Однією з переваг семантичного пошуку є його здатність розуміти значення та контекст запиту, а не лише знаходити збіги за ключовими словами чи метаданими. Завдяки цьому технологія дозволяє отримувати більш точні та релевантні результати, навіть якщо формулювання запиту відрізняється від опису зображення або не містить жодного з ключових термінів. Зазвичай, для досягнення цього використовується технологія K-Nearest Neighbors (KNN).

KNN (або К-найближчий алгоритм) – це контрольований алгоритм навчання, який працює за принципом, згідно з яким кожна точка даних, розташована поруч одна з одною, відноситься до одного класу. Основне припущення тут полягає в тому, що речі, які знаходяться поруч одна з одною, схожі одна на одну (рисунок 1.2). Здебільшого використовується алгоритм KNN через його легкість інтерпретації та короткий час обчислення [6].

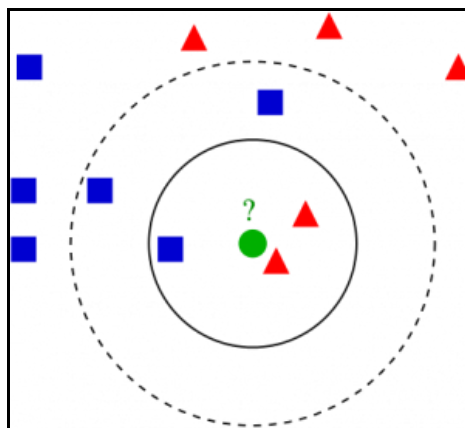


Рисунок 1.2 – Візуалізація даних на графіку [7]

На зображеному графіку можна побачити групу точок даних, розподілених невеликими кластерами. KNN перевіряє розподіл точок даних і, залежно від аргументів, наданих моделі, розділяє точки даних на групи. Потім цим групам присвоюється мітка. Основне припущення, яке робить модель KNN, полягає в тому, що точки даних/екземпляри, які існують у безпосередній близькості одна до одної, дуже подібні, тоді як якщо точка даних знаходиться далеко від іншої групи, вона не схожа на ці точки даних.

Модель KNN обчислює подібність, використовуючи відстань між двома точками на графіку. Чим більша відстань між точками, тим менше вони схожі. Існує кілька способів обчислення відстані між точками, але найпоширенішим показником відстані є просто евклідова відстань (відстань між двома точками на прямій лінії).

Векторний пошук – це метод, що використовується в інформаційному пошуку та машинному навчанні для швидкого знаходження елементів у

великому наборі даних. Він працює шляхом зберігання та групування елементів на основі їхніх векторних перетворень, в векторній базі даних. Ці перетворення є послідовностями чисел, які відповідають атрибутам елемента, незалежно від того, чи є цей елемент словом, документом, зображенням, аудіо- чи відеофайлом [8].

Векторний пошук перетворює дані, на числові вектори, які розташовуються у багатовимірному просторі. Це дозволяє знаходити інформацію не за точним збігом слів, а за їхнім семантичним змістом, використовуючи міри схожості.

Наприклад, для ефективної векторизації даних може бути використано мультимодальні моделі, такі як CLIP, що поєднують обробку зображень і тексту в єдиному латентному просторі. Цю модель можливо донавчати на специфічних даних з метою покращення представлення ознак об'єктів, які користувач прагне зберігати або шукати. Це сприяє реалізації семантичного пошуку, який демонструє вищу релевантність результатів порівняно з класичними методами. Додатково, модель BLIP дозволяє не лише узгоджувати зображення з текстовими описами, а й генерувати нові підписані зображення, що значно розширює можливості побудови мультимодальних векторних подань. Використання BLIP у поєднанні з CLIP забезпечує більш гнучке та контекстно-залежне моделювання даних. Векторні подання, отримані за допомогою таких моделей, можуть бути ефективно збережені у векторних базах даних, що оптимізує процеси пошуку та подальшої аналітики.

Векторна база даних – це сховище даних, поданих у вигляді математичних векторів. Вони значно спрощують для моделей машинного навчання запам'ятовування попередніх входів, що дозволяє використовувати машинне навчання для реалізації функцій пошуку, рекомендацій та генерації тексту. Завдяки векторним базам даних, дані можуть бути ідентифіковані на основі метрик подібності, а не лише точних збігів, що дає комп'ютерній моделі змогу розуміти дані в контексті [9].

Запит за вмістом зображення або пошук візуальної інформації на основі вмісту – це техніка, яка використовує алгоритми комп’ютерного зору для пошуку зображень на основі їх візуального вмісту. На відміну від традиційних методів пошуку зображень на основі тексту, які покладаються на ключові слова або теги, даний метод безпосередньо аналізує візуальні характеристики зображень, такі як колір, текстура, форма та об’єкти. Це забезпечує більш інтуїтивно зрозумілий і точний пошук зображень, особливо у випадках, коли ключові слова можуть бути нечіткими або неоднозначними [10].

CBIR має широке застосування у різних сферах. У пошукових системах ця технологія дозволяє отримувати більш точні та релевантні результати зображень. У медичній сфері CBIR допомагає в аналізі та діагностиці медичних знімків, таких як рентгенівські знімки, комп’ютерна томографія або магнітно-резонансна томографія. У галузі мистецтва та культурної спадщини вона застосовується для класифікації, організації та пошуку творів мистецтва, артефактів і історичних документів.

Комп’ютерний зір відіграє ключову роль у системах CBIR, оскільки забезпечує основні алгоритми та методи для вилучення ознак і вимірювання подібності [11]. Досягнення у сфері глибокого навчання, зокрема використання згорткових нейронних мереж, значно підвищили точність та ефективність сучасних систем CBIR (рисунок 1.3).

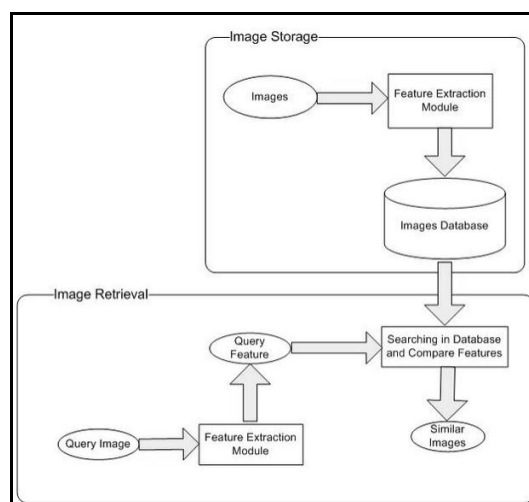


Рисунок 1.3 – Робочий потік CBIR системи [12]

Отже, інтелектуальні методи підбору зображень, такі як семантичний пошук, векторний пошук, пошук за вмістом зображення, гібридні підходи забезпечують ефективне виявлення більш релевантних результатів, порівняно з традиційними текстовими пошуковими системами. Застосування моделей CLIP та VLIP, а також векторних баз даних дозволить покращити процес автоматизованого підбору релевантних зображень на основі семантичного пошуку.

### **1.3 Аналіз наукових праць та існуючих програмних засобів для підбору релевантних зображень**

Пошук релевантних зображень є актуальною темою, як з точки зору досліджень, так і з точки зору комерційних застосувань.

У дослідженні [13] автори аналізують поведінку користувачів під час візуального та текстового пошуку в мобільному додатку електронної комерції. На основі логів запитів було виявлено, що гібридні запити (візуальний + текстовий) демонструють найвищий рівень конверсії – 3.5%, порівняно з 2.8% для текстових і 1.7% для візуальних запитів. Було також встановлено, що сесії з візуальним пошуком коротші, а кліків – менше, ніж у текстовому пошуку. Автори зазначають, що візуальний пошук частіше застосовується для категорій товарів, де зовнішній вигляд є критичним (наприклад, одяг). Дослідження також пропонує моделі передбачення для оцінки якості запитів і підкреслює потенціал гібридного пошуку для покращення користувацького досвіду. Також, автори дослідили, який відсоток від всіх запитів, становить візуальний пошук, протягом дня (рисунок 1.4).

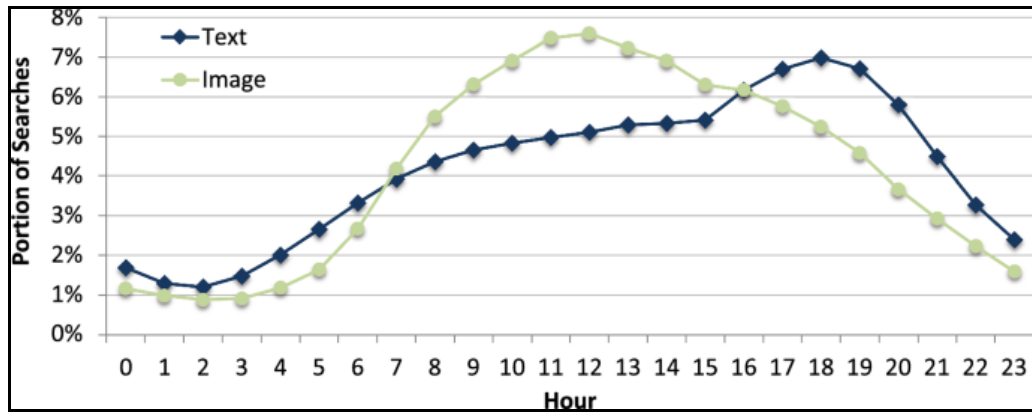


Рисунок 1.4 – Графік порівняння пошуків за картинкою та пошуків за текстом, протягом дня [13]

Автори [14] досліджують ефективність пошуку зображень за вмістом, шляхом розробки гібридного двошарового підходу, який поєднує локальні та глобальні (форма, текстура, колір) ознаки зображень. Перший шар виключає несхожі зображення, тоді як другий шар детально порівнює відібрані результати. Автори також запроваджують динамічну генерацію сценаріїв пошуку для підвищення точності. Експерименти на наборі даних Corel-1K демонструють підвищення точності до 86,06% для top-10 та 80,72% для top-20 результатів, що перевищує показники сучасних методів.

У дослідженні [15] автори вивчають роль візуального пошуку у сфері культурної спадщини, зокрема в рамках цифрових архівів історичних фотографій. Вони аналізують, як 15 експертів здійснюють пошук зображень для дослідницьких і творчих завдань, і збирають 37 реальних прикладів пошукових запитів, а також 158 зафіксованих бар'єрів. Було встановлено, що користувачі комбінують різні тактики (ключові слова, фільтрацію, перегляд), проте основною перешкодою виявився низький рівень метаданих, що обмежує ефективність пошуку за ключовими словами. Автори підкреслюють, що наявність цифрових архівів недостатня без зручних механізмів доступу до вмісту. У дослідженні також акцентується на потенціалі автоматичної анотації для покращення пошуку, а також необхідності глибшого розуміння взаємодії людини з візуальними даними в контексті цифрових гуманітарних наук.

Зворотний пошук зображень Google, офіційно названий Google Search by Image – це служба Google, яка дозволяє користувачеві шукати зображення, використовуючи зображення як початкову точку, а не письмовий чи усний пошуковий запит [16].

Користувач завантажує зображення або надає посилання на зображення, яке можна знайти в Інтернеті, і Google спробує знайти відповідні зображення. Зазвичай це будуть схожі зображення або суміш схожих зображень і точних копій (рисунок 1.5).

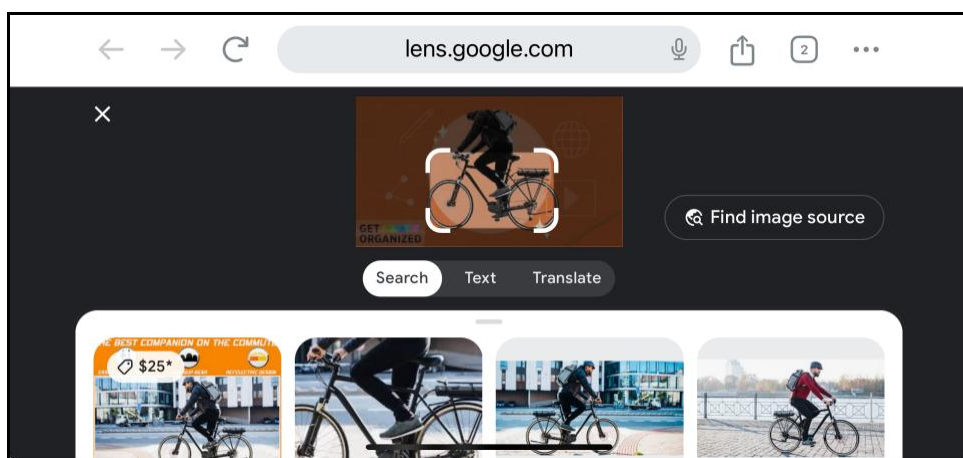


Рисунок 1.5 – Приклад роботи Google Search by Image [17]

Зображення, яке використовується для зворотного пошуку, аналізується на наявність точок, кольорів, ліній і текстур. Запит, створений на основі цього аналізу, потім порівнюється з мільярдами можливих зображень, до яких Google має доступ. Якщо буде знайдено точний збіг, Google відобразить це, вказавши різні розміри зображень, які вдалося знайти. Зазвичай також відображаються пов'язані зображення разом із відповідними веб-сторінками.

Google також пропонує застосунок на Android та iOS – Google Lens, який також надає результати пошуку на основі зображень, хоча він також може сканувати QR-коди, перекладати текст тощо.

Різниця між Google Lens і стандартною функцією зворотного пошуку зображень полягає в тому, що по-перше, користувач можете використовувати Google Lens у режимі реального часу, наприклад під час перекладу тексту.

Користувачу навіть не потрібно робити знімок, достатньо навести на нього камеру смартфона, будь то дорожній знак чи меню в ресторані.

Втім, Google Image Search має й декілька суттєвих недоліків. Насамперед, обмежене розуміння намірів користувача. Система не завжди може визначити, що саме необхідно шукати: схожий об'єкт, бренд, стиль чи джерело зображення. Як наслідок, це призводить до поверхневого семантичного розпізнавання – алгоритми здатні визначити загальні, але не завжди ідентифікують конкретну модель, бренд чи стиль. Крім того, результати пошуку іноді є нерелевантними або випадковими, оскільки система може показувати просто популярні або добре проіндексовані зображення, а не ті, що реально відповідають запиту.

Alibaba – це глобальний онлайн-ринок, який об'єднує покупців і продавців, пропонуючи широкий спектр товарів і послуг, від електроніки до одягу, зі зручним інтерфейсом і безпечними варіантами оплати. Функція пошуку зображень продуктів Alibaba дозволяє користувачам шукати продукти за зображенням, а не за ключовими словами чи описами. Цей інструмент є зручним способом знайти схожі товари, знайти виробників і порівняти ціни на основі візуальних посилань [18].

Alibaba Cloud Image Search – це інтелектуальний сервіс зображень, заснований на машинному та глибокому навчанні. Він інтегрує пошук і розпізнавання зображень, що дозволяє користувачам шукати зображення в різних галузях і бізнес-сценаріях. Він спрямований на надання користувачам ідентичних або подібних зображень (рисунок 1.6).

Щоб покращити пошук товарів в Інтернеті, Alibaba Cloud інтегрувала пошук зображень у програму Taobao у формі функції Pailitao. З моменту випуску Pailitao у 2014 році кількість унікальних щоденних відвідувачів постійно зростає. Це число зросло з кількох сотень до понад десяти мільйонів на день. Зростання передбачає велике впровадження користувачами функції пошуку за зображенням для онлайн-покупок. На відміну від традиційного пошуку за ключовими словами, Pailitao вимагає лише простої фотографії продукту без

будь-якого текстового опису продукту. Це робить процедуру онлайн-покупок зручною та покращує взаємодію з користувачем [19].

Alibaba Cloud Image Search має ряд переваг, які полегшують життя користувача. Завдяки Alibaba Cloud MaxCompute, Alibaba Cloud Image Search підтримує понад десять мільярдів елементів. Це гарантує, що пошук зображень обслуговує компанії, які належать до повного спектру галузей, типів, розмірів і географічних регіонів.

Також різні організації мають унікальні вимоги. Такі фактори, як тип організації, цільовий ринок і розмір ринку, разом визначають вимоги до пошуку зображень. Alibaba Cloud Image Search пропонує гнучкість для задоволення індивідуальних вимог компаній.



Рисунок 1.6 – Приклад роботи реверсивного пошуку зображень в Таобао [20]

Втім, сервіс має й деякі недоліки. Головна проблема – погане розуміння контексту. Наприклад, система не розрізняє, чи хоче людина знайти точний товар, його дешевший аналог, тільки аксесуар із зображення або просто дізнатися назву предмета. Ще один недолік – труднощі з багатокomпонентними зображеннями. Якщо на фото зображено кілька об'єктів, система не завжди правильно вибирає, на що орієнтуватися. До того ж вона досить чутлива до

якості зображення: розмиті, затемнені або частково обрізані фото можуть значно знизити точність пошуку.

Отже, згідно з проведеного аналізу відомих наукових досліджень та програмних рішень, пошук релевантних зображень це популярна задача, для розв'язання якої використовуються різні підходи проте доволі обмежено, за рахунок необхідності значних обчислювальних ресурсів, та часто без розуміння контексту наданого для пошуку зображення. Тому є доцільним покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних.

#### **1.4 Мета та задачі кваліфікаційної роботи бакалавра**

Метою кваліфікаційної роботи є покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних.

Для досягнення мети, ставляться такі задачі:

- виконати дослідження предметної області для задачі пошуку релевантних зображень;
- розробити метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних;
- виконати програмну реалізацію створеного методу автоматизованого підбору релевантних зображень на основі семантичного пошуку;
- провести дослідження роботи методу з використанням розробленої інтелектуальної системи.

## Розділ 2 Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

### 2.1 Спосіб отримання нейромережевої моделі для реалізації семантичного пошуку

Для покращення процесу автоматизованого підбору релевантних зображень шляхом семантичного пошуку, нейромережеву модель, що використовуватиметься для векторизації даних, необхідно донавчити.

Схему донавчання моделі векторизації для семантичного пошуку наведено на рисунку 2.1.

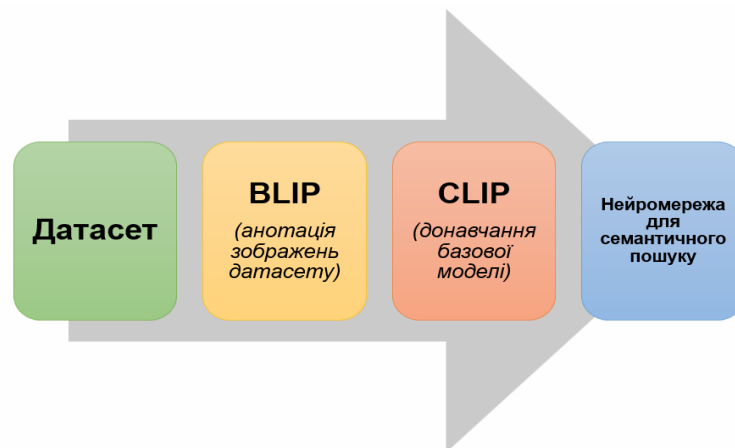


Рисунок 2.1 – Донавчання моделі для семантичного пошуку

На першому етапі здійснюється анотація зображень, які будуть використані, як основа для донавчання системи. Оскільки більшість датасетів не є анотованими, для них необхідно згенерувати текстові описи, які виступатимуть смисловими маркерами для подальшого навчання. Для автоматизації цього процесу застосовується спеціалізована модель BLIP. Це суттєво скорочує обсяги ручної роботи та забезпечує більшу уніфікованість підписів.

Після формування пар «зображення – опис» на другому етапі виконується донавчання базової моделі CLIP. Вона має навчитись встановлювати відповідність між зображеннями та їх описами та, пов'язувати

між собою подібні зображення схожими векторами, за рахунок чого можна виконати пошук релевантних зображень.

Отже, після виконання даних етапів, буде отримано нейромережеву модель, яка забезпечує векторизацію та пошук релевантних зображень. Такий підхід дозволить покращити процес автоматизованого підбору релевантних зображень.

## 2.2 Схеми та кроки методу автоматизованого підбору релевантних зображень засобами векторних баз даних

Для покращення процесу автоматизованого підбору релевантних зображень запропоновано метод, який дозволяє знайти, релевантні до запиту користувача, зображення. На рисунку 2.2 зображено схему даного методу.

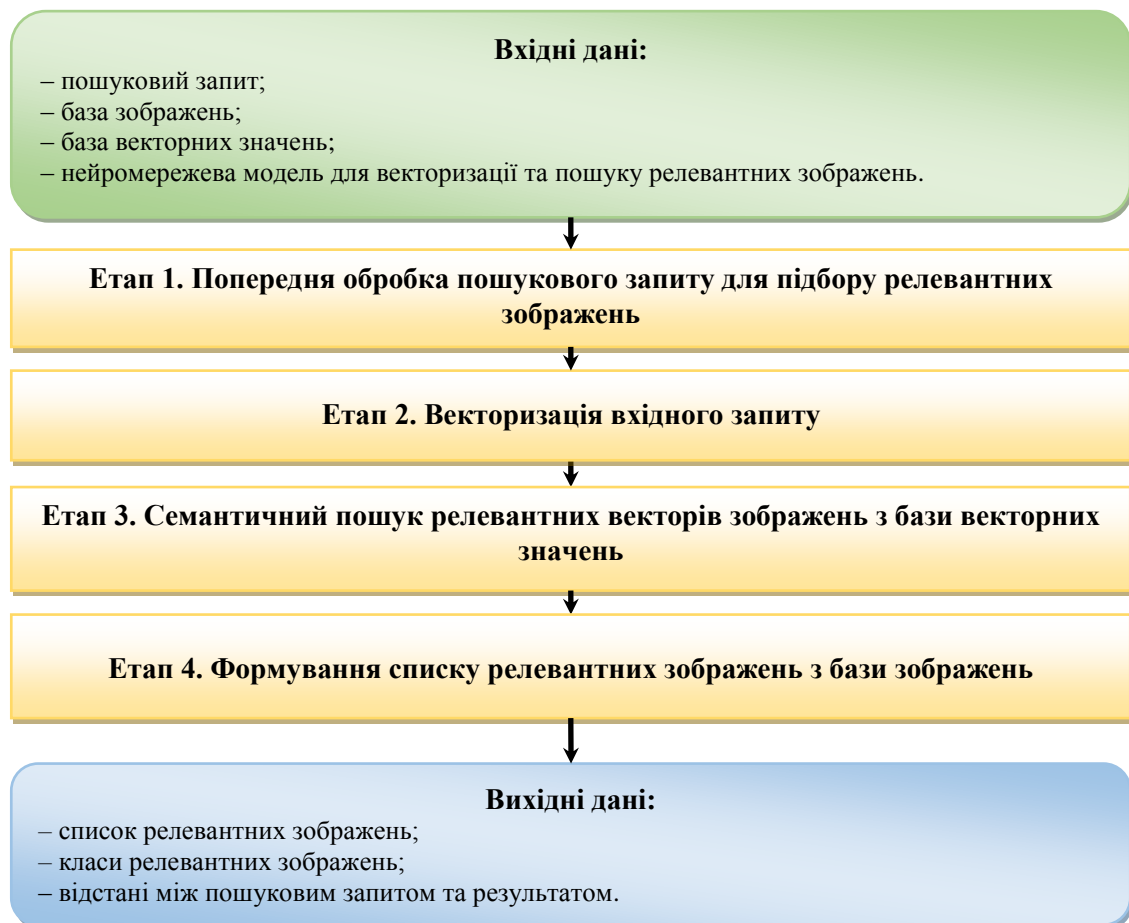


Рисунок 2.2 – Схема методу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

Вхідними даними методу є пошуковий запит (текст або зображення), база зображень, база векторних значень та нейромережева модель для векторизації та порівняння векторів.

На етапі 1 попередньої обробки для текстових запитів здійснюється лінгвістичне очищення та кодування, а для зображень – перетворення до стандартного формату, що підтримується моделлю (а саме – RGB). Така уніфікація даних дозволяє забезпечити коректну подачу інформації на вхід нейромережевої моделі, яка працює як із текстом, так і з візуальними об'єктами.

Після цього на етапі 2 підготовлений запит перетворюється у векторне значення за допомогою попередньо натренованої мультимодальної нейромережі CLIP, процес донавчання якої було описано у пункті 2.1. Дане векторне значення фіксує змістовні особливості зображення або тексту у багатовимірному просторі. Вектори однаково репрезентують і текстові, і візуальні об'єкти, що дозволяє виконувати кросмодальний пошук. Завдяки цьому етапу пошуковий запит можна порівняти з елементами бази даних у єдиному векторному просторі.

На 3-му етапі отриманий вектор запиту використовується для знаходження найближчих векторів у базі векторних значень. Семантичний пошук реалізується через механізм k-найближчих сусідів. Векторна БД виконує пошук, щоб ідентифікувати «k» векторів у своїй колекції, які є найближчими до вектора запиту. Цей процес ґрунтується на розрахунку відстані/подібності між векторами. У цьому етапі враховується не лише візуальна схожість, а й глибші семантичні зв'язки, що дає змогу знайти зображення з подібним змістом навіть за умов відсутності візуальної ідентичності. Результатом є список найближчих векторів, які зберігають найвищий ступінь релевантності.

Після ідентифікації релевантних векторів, на етапі 4 отримується шлях до збережених зображень, які пов'язані з цими векторами. Формується підсумковий список зображень, який містить також інформацію про клас зображення всередині бази даних та відстань його вектору, до вектору запиту. Завдяки цьому користувач може отримати не лише релевантні зображення, а й оцінити ступінь

їх відповідності початковому запиту. Таким чином, завершується процес семантичного пошуку.

Вихідними даними методу є перелік зображень, релевантних до запиту. Разом з ними відображаються додаткова інформація про результати.

Отже, запропонований у пункті метод автоматизованого підбору релевантних зображень засобами векторних баз даних дозволить покращити процес автоматизованого підбору релевантних зображень за рахунок використання донавченої нейромережевої моделі CLIP.

### 2.3 Проектна архітектура та взаємозв'язок компонентів інтелектуальної системи автоматизованого підбору релевантних зображень

Згідно поставлених завдань виникає необхідність спроектувати архітектуру інтелектуальної системи автоматизованого підбору релевантних зображень. Проектна архітектура системи показана на рисунку 2.3.

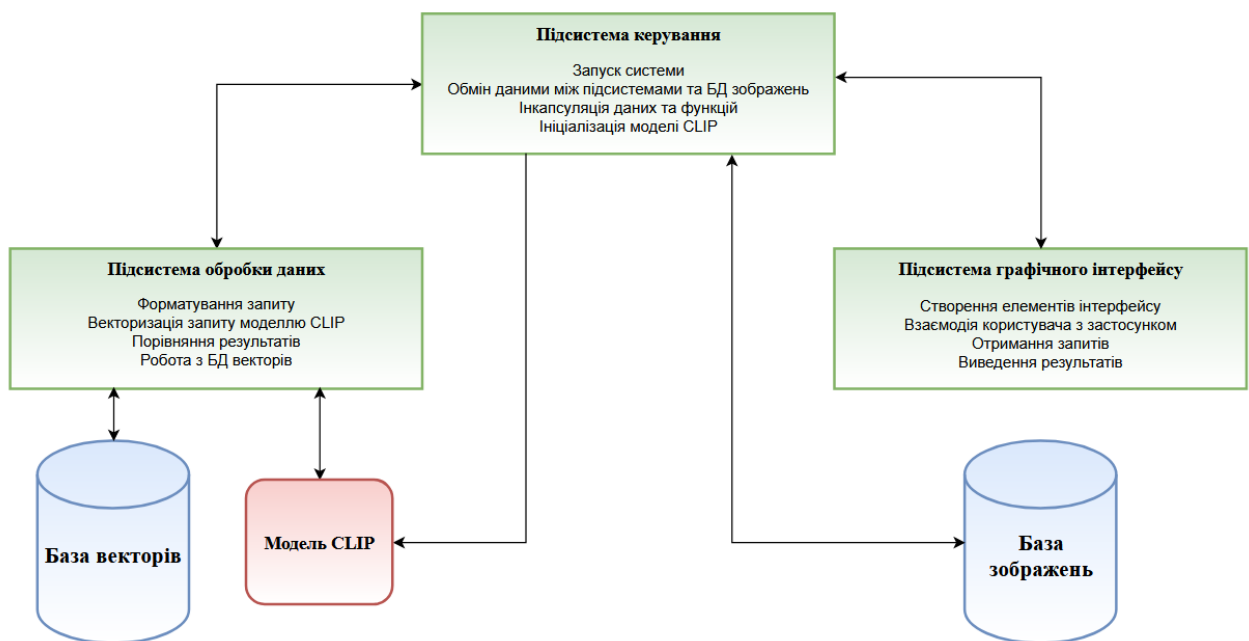


Рисунок 2.3 – Проектна архітектура

Інтелектуальна система автоматизованого підбору релевантних зображень складається з трьох підсистем: підсистеми керування, підсистеми

графічного інтерфейсу та підсистеми обробки даних. Дані підсистеми взаємодіють з моделлю векторизації CLIP та двома базами даних: бази векторів та бази зображень.

Підсистема керування є центральним компонентом інтелектуальної системи, що відповідає за початковий запуск системи, обмін даними між підсистемами та базою зображень, інкапсуляцію даних і функцій, а також ініціалізацію моделі CLIP. Вона обробляє дії користувача, викликає функції для відповідних пошукових запитів, виступаючи як єдина точка взаємодії для користувача з внутрішньою реалізацією системи.

Підсистема графічного інтерфейсу відповідає за створення елементів інтерфейсу, таких як вікна, кнопки та форми введення. Вона забезпечує взаємодію користувача із застосунком, отримуючи запити від користувача та відображає результати пошуку, отримані від підсистеми керування.

Підсистема обробки даних реалізує основну функцію застосунку – пошук релевантних зображень. Вона приймає на вхід запити, форматує їх, векторизує запит за допомогою моделі CLIP, порівнює отримані результати та взаємодіє з базою векторів для пошуку схожих значень.

База векторів зберігає векторні значення, які використовуються підсистемою обробки даних для пошуку схожих зображень, в той час, як база зображень зберігає самі зображення, які підсистема керування використовує для виведення результатів, зіставляючи знайдені вектори з відповідними зображеннями.

Для кращого розуміння процесів, що відбуватимуться в спроектованій системі було описано діаграму послідовності та діяльності.

Діаграма послідовності – це діаграма, яка використовується для моделювання логіки сценаріїв використання, показуючи інформацію, що передається між об'єктами в системі під час виконання сценарію [21].

На рисунку 2.4 наведена діаграма послідовностей інтелектуальної системи автоматизованого підбору релевантних зображень.

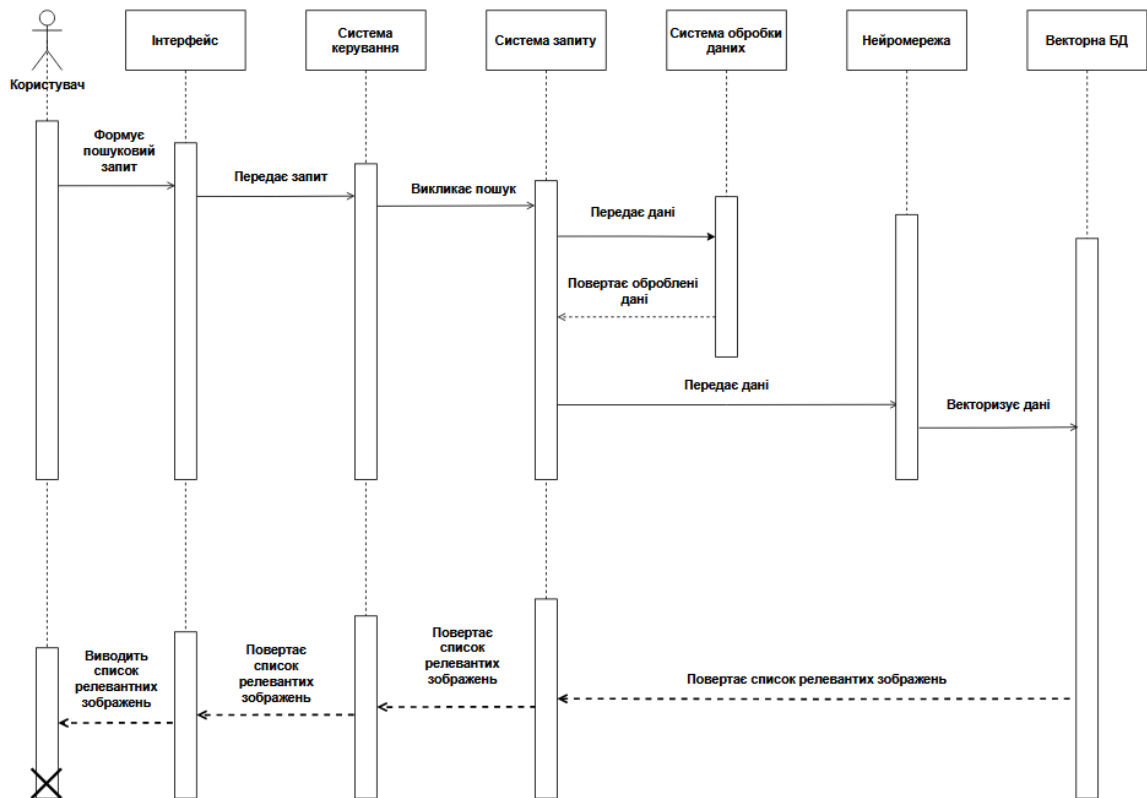


Рисунок 2.4 – Діаграма послідовності

Користувач ініціює взаємодію із системою, формуючи пошуковий запит через інтерфейс. Цей запит може бути як текстовим, так і візуальним.

Далі, запит передається до системи керування, яка відповідає за координацію дій усіх складових системи, включно з переміщенням даних між графічним інтерфейсом та системою запитів.

Коли система керування отримує дані від графічного інтерфейсу, вона передає ці дані далі, до системи запиту, яка ініціалізує процес пошуку. Але, перед цим, система запиту надсилає дані до системи обробки даних для попередньої підготовки. Система обробки аналізує отриману інформацію та повертає оброблені дані назад.

Після цього, система запиту надсилає оброблені дані до неймережі, яка векторизує запит. Цей вектор передається у векторну базу даних, де виконується пошук за схожими векторами.

Після завершення пошуку векторна база даних повертає результати назад до користувача, в наступній послідовності: «від БД до системи запиту, від

системи запиту до системи керування, від системи керування до графічного інтерфейсу». Таким чином, користувач отримує відповідь на свій запит.

Діаграми діяльності описують, як дії координуються для виконання задачі. Зазвичай для досягнення певної події необхідно виконати низку операцій, особливо коли операція спрямована на досягнення кількох цілей, що потребують координації, або коли події в межах одного випадку використання пов'язані між собою [22].

На рисунку 2.5 наведено діаграму активності інтелектуальної системи автоматизованого підбору релевантних зображень.

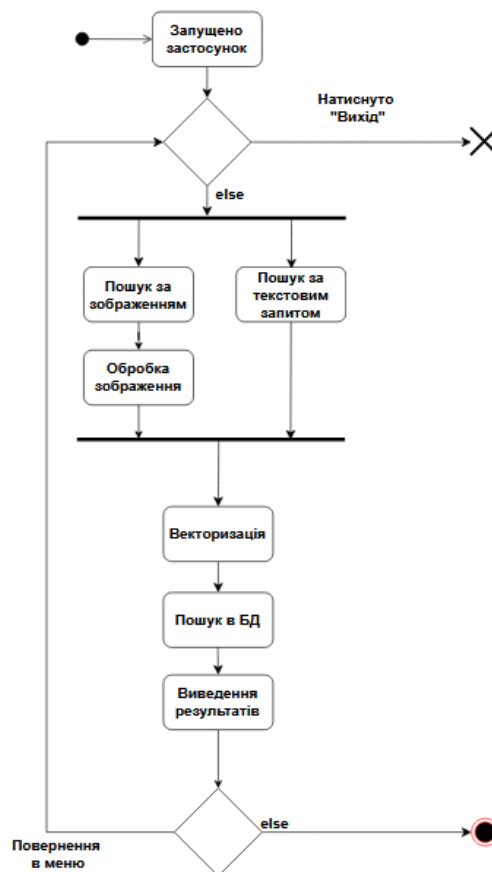


Рисунок 2.5 – Діаграма активності

Початковою точкою застосунку є його запуск. Після запуску система дозволяє користувачу прийняти рішення, в головному меню: або користувач натисне на вихід, що завершить роботу системи, або продовжить роботу в застосунку.

Після розгалуження процес розділяється на дві паралельні активності, які можуть бути виконані користувачем: «Пошук за зображенням» та «Пошук за текстовим запитом». Таким чином, система може виконувати пошук інформації, використовуючи або зображення, або текстовий запит як вхідні дані. Якщо обрано «Пошук за зображенням», за ним слідує «Обробка зображення», що є необхідним кроком перед подальшою роботою застосунку.

Після об'єднання цих паралельних процесів виконується «Векторизація», що, означає перетворення вхідних даних у векторний формат для подальшого їх використання. Це дозволяє виконати «Пошук в БД» та виведення даних для користувача, як результат. Далі користувач може або повернутись в меню і повторити процес, або завершити роботу з застосунком, оскільки виконав поставлену задачу.

Отже, в даному розділі було описано проєктну архітектуру та взаємозв'язок між компонентами інтелектуальної системи автоматизованого підбору релевантних зображень. Розробка такої інтелектуальної системи дозволить вирішити поставлені у пункті 1.4 завдання, а саме виконати програмну реалізацію створеного методу та провести його експериментальне дослідження.

## **2.4 Схема та кроки процесів векторизації та анотації даних**

Одним із засобів комп'ютерного зору та обробки природної мови є модель CLIP (Contrastive Language-Image Pretraining), розроблена компанією OpenAI. CLIP є універсальним засобом векторизації, який поєднує можливості нейронних мереж для зображень та тексту, дозволяючи здійснювати ефективне зіставлення зображень і текстових описів у спільному векторному просторі. Це дає змогу реалізовувати завдання візуального розпізнавання без потреби в переорієнтації моделі на конкретні набори даних [23].

CLIP навчається за принципом контрастного навчання: параметрам «зображення – відповідний текстовий опис» надається висока схожість векторних значень, тоді як невідповідним параметрам – низька. Такий підхід

дозволяє моделі формувати узагальнене розуміння зображень на основі мовного контексту, що суттєво розширює сферу її застосування. Архітектура моделі CLIP наведена на рисунку 2.6.

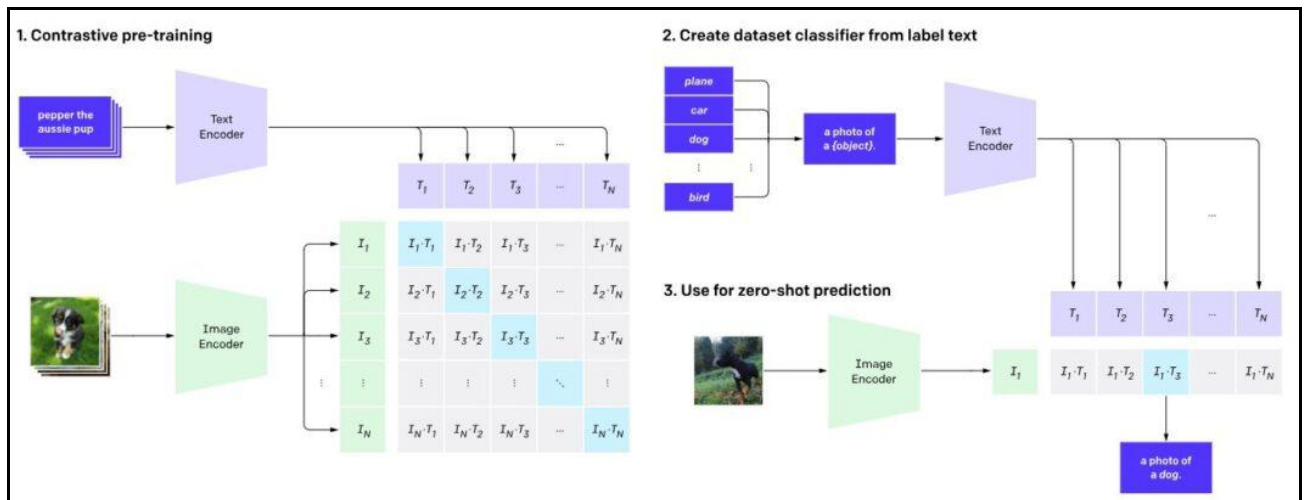


Рисунок 2.6 – Архітектура моделі CLIP [24]

Під час навчання CLIP обробляє сотні мільйонів зображень із супровідними підписами, зібраних із відкритих джерел. Це забезпечує моделі високу здатність до генералізації. Кінцевим результатом є можливість застосування CLIP до нових завдань «нульового пострілу» (zero-shot learning) – тобто без додаткового навчання на конкретних прикладах.

CLIP складається з двох основних компонентів: візуального енкодера і текстового енкодера. Обидві частини перетворюють відповідні вхідні дані (зображення або текст) у вектори фіксованої розмірності. Потім ці вектори порівнюються за косинусною схожістю, що дозволяє визначити рівень їх відповідності [25].

CLIP є ефективним засобом векторизації, оскільки генерує щільні векторні значення зображень і текстів, які зберігають семантичну подібність. Це дозволяє використовувати модель для швидкого пошуку та кластеризації.

Оскільки семантичний пошук розробленого методу працює із гібридним пошуковим запитом, що включає як текстовий опис, так і зображення, для донавчання моделі, процес якого наведено у пункті 2.1, обраний датасет

необхідно проанотувати, оскільки він не містить описової частини. Для виконання цієї задачі було використано модель BLIP.

BLIP [26] (Bootstrapping Language-Image Pre-training) – це модель, розроблена компанією Hugging Face, яка покликана об'єднати сфери обробки природної мови та комп'ютерного зору. Завдяки масштабному попередньому навчанню на мільйонах пар «зображення–текст», BLIP ефективно виконує поставлені перед ним завдання, зокрема автоматичне створення підписів до зображень, візуальне запитання-відповідь, крос-модальний пошук [27]. Архітектуру BLIP наведено на рисунку 2.7.

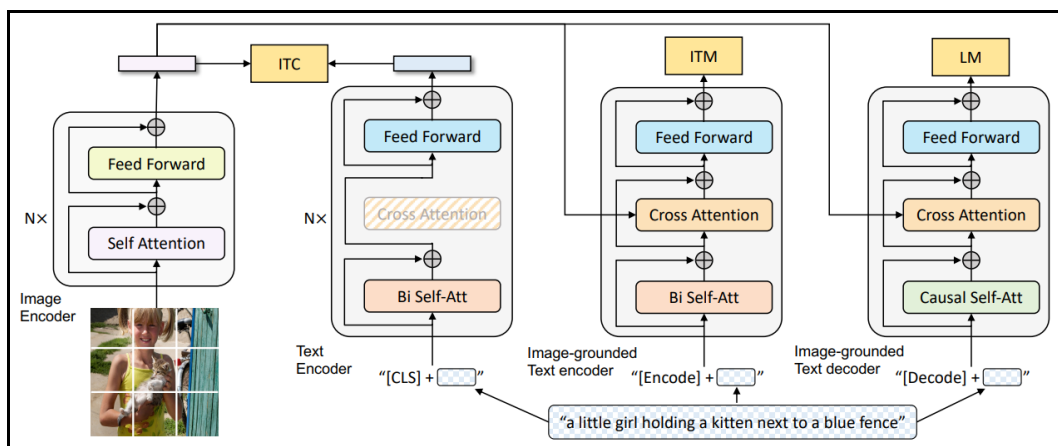


Рисунок 2.7 – Архітектура BLIP [28]

Архітектура BLIP побудована для ефективного поєднання зображення й тексту у єдиному мультимодальному просторі. Вона складається з декількох компонентів, які забезпечують як розуміння, так і генерацію мультимодальних даних.

Першим етапом є унімодальне кодування. Зображення обробляються візуальним енкодером, який перетворює зображення в набір векторів ознак. Паралельно текст проходить через текстовий енкодер, який кодує текстову інформацію в окремий набір векторних значень. На цьому етапі кожна модальність опрацьовується незалежно.

Наступним кроком є злиття модальностей. BLIP використовує текстовий енкодер, який дозволяє текстовим анотаціям враховувати візуальний контекст.

Це означає, що модель може «зосередитись» на відповідних частинах зображення, щоб краще зрозуміти значення тексту.

Для генеративних завдань, таких як створення підписів до зображень, VLP має текстовий декодер із прив'язкою до зображення. Він працює автогенеративно, по черзі генеруючи слова підпису, при цьому постійно враховуючи візуальні ознаки. Так забезпечується точність та зв'язність створених описів.

Отже, було описано архітектури засобу векторизації зображень, використаного при розробці – CLIP, а також засобу анотації зображень – VLP. Дані моделі ефективні для об'єднання мовних і візуальних елементів у єдиному просторі. Використання наведених засобів дозволить реалізувати наведений розділі метод.

## **2.5 Датасет для донавчання нейромережевої моделі**

Датасет у машинному навчанні та штучному інтелекті – це сукупність даних, яка використовується для навчання та тестування алгоритмів і моделей. Такі набори є ключовими для розробки та успішного функціонування систем машинного навчання й штучного інтелекту, оскільки саме вони надають необхідні вхідні та вихідні дані, на основі яких алгоритми можуть навчатися [29].

У машинному навчанні можуть застосовуватися набори даних різних типів, як структуровані, так і неструктуровані дані. Структурованими називають дані, організовані у певний спосіб, наприклад, у вигляді електронних таблиць або баз даних. Такий тип даних легко піддається аналізу, оскільки вже перетворений до зручного формату. Неструктуровані дані, навпаки, не мають визначеної структури – це може бути текст, зображення тощо. Перед використанням такі дані потребують додаткової обробки та аналізу.

Для анотації та векторизації даних з метою донавчання CLIP було обрано датасет Caltech101 [30]. Даний набір даних містить зображення зі 101 категорії

об'єктів і фонову категорію, яка містить зображення, що не належать до 101 категорії об'єктів (рисунок 2.8). Для кожної категорії об'єктів існує від 40 до 800 зображень, тоді як більшість класів мають близько 50 зображень. Роздільна здатність кожного зображення становить  $300 \times 200$  пікселів [31].

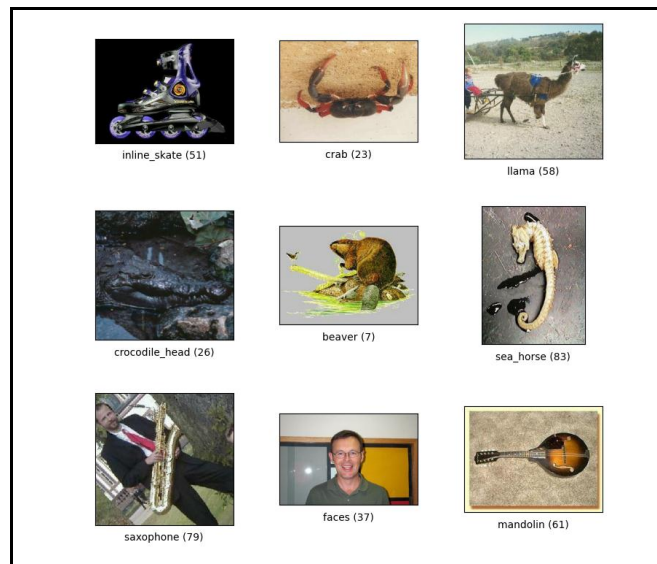


Рисунок 2.8 – Приклад змісту датасету [32]

Зображення з кожної категорії об'єктів були зафіксовані за різних умов освітлення, фону та точок огляду, що робить набір даних хорошим кандидатом для навчання надійної моделі комп'ютерного зору.

Нижче на рисунку 2.9 наведено діаграму, яка показує розподілення зразків датасету по найбільших класах.

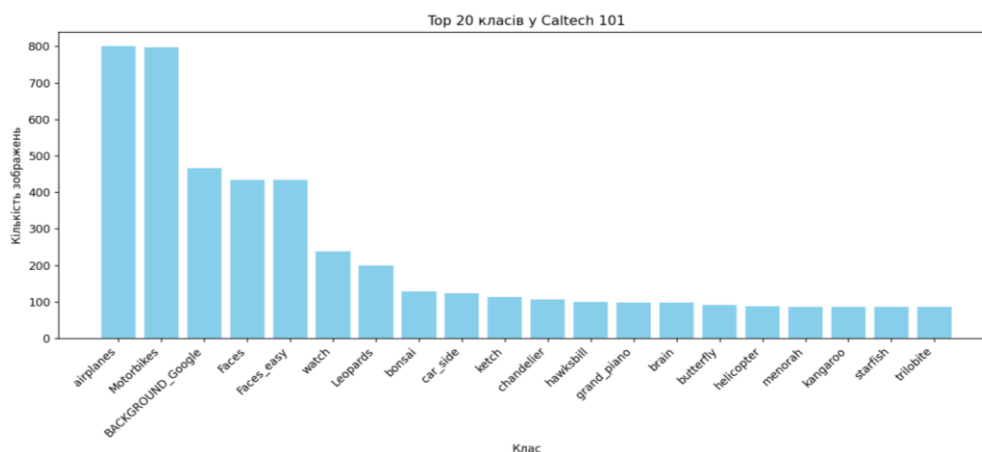


Рисунок 2.9 – Графік розподілу зразків у найбільших 20 класах набору даних

Використання набору даних Caltech 101 має низку переваг у порівнянні з іншими наборами для розпізнавання об'єктів. Зокрема, зображення всередині кожної категорії мають однаковий розмір та подібне розташування об'єктів, що полегшує процес навчання моделей. Крім того, набір відзначається низьким рівнем шуму та оклюзії, що дозволяє алгоритмам зосереджуватись на унікальних ознаках об'єктів [33].

Оскільки набір даних спочатку не був векторизований, виникла потреба у його попередній обробці перед використанням нейромережею. Для забезпечення ефективної роботи моделі з цими зображеннями датасет було векторизовано спеціально розробленим застосунком – кожне зображення перетворено на числове значення у вигляді векторів ознак. Така трансформація дозволила надалі використовувати набір для підбору релевантних зображень.

Отже, з метою донавчання нейромережі для семантичного пошуку CLIP було обрано датасет Caltech101. Для виконання кроків методу автоматизованого підбору релевантних зображень засобами векторних баз даних датасет був анотований за допомогою VLIP та векторизований.

## **2.6 Статистичні метрики**

Дослідження донавченої нейромережевої моделі на етапі 3 запропонованого методу необхідно розглянути метрики, що будуть використовуватись для оцінки даної моделі. При проведенні дослідження було використано такі метрики, як: середнє значення функції втрат, асигасу, косинусна схожість.

Середнє значення втрат – це функція, яка використовується для задач виявлення об'єктів та сегментації екземплярів. Якщо обрати середнє значення втрат, як один із параметрів експерименту, то після завершення навчання можна отримати графік, що демонструє, як змінювалося значення втрат протягом усього процесу навчання [34].

Після кожного передбачення модель порівнює свої результати з правильними мітками, за допомогою функції втрат. Для кожного прикладу в наборі даних обчислюється величина помилки. Усі значення втрат підсумовуються і діляться на кількість прикладів, щоб отримати середню помилку (середнє значення втрат) за одну епоху. Таким чином, середнє значення втрат показує, наскільки добре модель загалом узгоджується з правильними відповідями на поточному етапі навчання. Зменшення цього значення протягом навчання свідчить про покращення якості моделі.

Метрика *accuracy* (top-1) стосується ефективності роботи класифікатора при виконанні завдання класифікації. Модель надає оцінку або рівень впевненості для кожного класу. Якщо основне передбачення класифікатора є коректним, то вважається, що було досягнуто очікуваної *accuracy* [35].

Зміна значення *accuracy* під час донавчання моделі CLIP свідчить про те, наскільки ефективно модель визначає єдину, найбільш правильну відповідність для кожної пари «зображення-текст». Збільшення даної метрики означає, що модель стає кращою у тому, щоб визначати правильну відповідь першою. Це прямий показник того, що модель успішно навчається вибудовувати сильний зв'язок між правильними парами зображень і текстів, роблячи їх найближчими у своєму внутрішньому представленні.

Косинусна схожість – це міра схожості між двома векторами у просторі скалярного добутку. Вона визначає ступінь, до якого два вектори спрямовані в одному напрямку, шляхом обчислення косинуса кута між ними. Косинусна схожість зазвичай використовується в аналізі, для вимірювання схожості між об'єктами на основі частоти елементів, які вони містять [36].

Для мультимодальних моделей, таких, як CLIP однією з ключових задач є створення векторних перетворень для даних різних модальностей, в даному випадку – зображення та тексту, у спільному просторі. Це дозволяє кількісно оцінювати семантичні зв'язки між об'єктами, незалежно від їхньої первинної модальності.

Метрика косинусної схожості дозволяє встановити, наскільки точно модель трансформує вхідні дані у вектори, тобто, чи семантично схожі об'єкти мають вектори з високим ступенем схожості та чи невідповідні об'єкти мають вектори, що характеризуються низькою схожістю або високою відстанню один від одного, у цьому спільному просторі.

Зростання косинусної схожості між векторними представленнями зображення та його відповідної текстової анотації, під час процесу навчання моделі, є індикатором успішної оптимізації та свідчить про ефективне вивчення моделлю необхідних семантичних кореляцій.

Косинусна схожість між двома векторами (Вектор А, Вектор В) обчислюється як скалярний добуток цих двох векторів, поділений на добуток їхніх довжин.

Необхідно кожен відповідну «координату» Вектора А помножити на відповідну «координату» Вектора В та додати всі ці добутки між собою. Цей добуток показує, наскільки «спрямовані» вектори в одному напрямку. Якщо вони спрямовані в одному напрямку, скалярний добуток буде додатним значенням, а якщо в протилежних – від'ємним.

Норма вектора – це дійсне число, яке узагальнює поняття «довжини» або «розміру» вектора. У звичайному 2D або 3D просторі норма відповідає інтуїтивній геометричній довжині вектора. В випадку прямокутного трикутника нормою буде довжина гіпотенузи.

Важливо знайти норм векторів, оскільки вони дозволяють «нормалізувати» результат. Тобто, гарантувати, що схожість залежить лише від напрямку векторів, а не від їхньої довжини. Це дозволяє порівнювати, наприклад, короткий і довгий текст, ігноруючи їхню абсолютну довжину та зосереджуючись на спільних темах або змісті.

Простими словами, це дозволяє отримати числове значення кута між двома векторами, таким чином порівнявши їх подібність. Висока косинусна схожість між явно схожими елементами (наприклад, синонімами) і низька між несхожими вказує на те, що модель згенерувала якісні, осмислені вбудовування.

Під час дотренування моделі, в ідеалі, вектори синонімів об'єктів, з якими працює і навчається система, мають наближуватись один до одного, наприклад, вектор зображення кота в полі та текст «кіт сидить в полі», за початкових налаштувань моделі будуть подібними, але не наближеними до ідентичних. Внаслідок дотренування модель може навчитись краще визначати подібність між ними, відповідно, надати їм більш наближені один до одного вектори, що в свою чергу, підвищить значення метрики косинусної схожості.

Отже, було описано метрики, що будуть використані для дослідження донавченої моделі векторизації CLIP, серед них середнє значення функції втрат, ассураcy top-1 та косинусна схожість.

## **2.7 Висновки до розділу 2**

У даному розділі запропоновано метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних, що дозволить підвищити якість процесу пошуку релевантних зображень.

Було розглянуто основні складові інтелектуальної системи автоматизованого підбору релевантних зображень. Наведено схему та кроки процесів векторизації та анотації даних. Розглянуто, як за допомогою методів векторизації тексту та подальшого порівняння векторів у багатовимірному просторі можна знаходити зображення, які є найбільш релевантними до запиту користувача.

Згідно поставлених завдань було спроектовано проєктну архітектуру інтелектуальної системи автоматизованого підбору релевантних зображень, яка передбачає наявність трьох основних підсистем, пов'язаних із векторною базою даних, моделлю CLIP та базою зображень. Наведено та описано діаграми активності та послідовності, що дозволило краще зрозуміти послідовність процесів у спроектованій інтелектуальній системі.

Крім того, було розглянуто датасет, використаний для роботи з інтелектуальної системи автоматизованого підбору релевантних зображень. Під

час роботи з даними обраного датасету було враховано особливості його структури, форматів збереження та необхідності попередньої обробки, зокрема засобами автоматичної векторизації та анотації: CLIP та BLIP, відповідно.

Для вирішення подальших задач, а саме програмної реалізації інтелектуальної системи автоматизованого підбору релевантних зображень та експериментального дослідження створеного методу з використанням необхідно обрати засоби реалізації інтелектуальної системи та спеціалізовані програмні компоненти, навести структуру розробленої інтелектуальної системи та виконати дослідження запропонованого у роботі методу.

## Розділ 3 Експериментальне дослідження

### 3.1 Засоби реалізації інтелектуальної системи автоматизованого підбору релевантних зображень

Вибір програмних засобів для реалізації інтелектуальної системи автоматизованого підбору релевантних зображень є важливим частиною розробки. Обрані технології визначають не лише точність і швидкість обробки даних, а й здатність системи масштабуватись, взаємодіяти з великими обсягами інформації та забезпечувати стабільну роботу у різних умовах використання. Для розробки було обрано наступні засоби: Visual Studio Code, як редактор коду, Python, як мову програмування, та ChromaDB, як базу даних.

Visual Studio Code [37] – це редактор вихідного коду, який працює як на комп'ютері, так і в браузері. Даний редактор доступний для Windows, macOS, Linux та Raspberry Pi OS. Він має вбудовану підтримку JavaScript, TypeScript і Node.js, а також багату екосистему розширень для інших мов програмування (таких як C++, C#, Java, Python, PHP та Go), середовищ виконання (таких як .NET і Unity), середовищ розгортання (таких як Docker і Kubernetes) та хмарних платформ (таких як Amazon Web Services, Microsoft Azure і Google Cloud Platform) [38].

Visual Studio Code побудована з використанням оболонки Electron, Node.js, TypeScript та протоколу Language Server Protocol, і оновлюється щомісяця. Рівень підтримки для різних мов програмування та їхніх розширень може змінюватися – від простого підсвічування синтаксису та парного виділення дужок до повноцінного рефакторингу.

Python [39] – це мова програмування високого рівня з широкими можливостями, відома своєю зрозумілістю та простотою. Вона підтримує кілька парадигм програмування, зокрема процедурне, об'єктно-орієнтоване та функціональне програмування [40].

Python використовується для розробки вебзастосунків на стороні сервера, створення програмного забезпечення, виконання математичних обчислень і

системного скриптингу. Вартість супроводу програм на Python знижується завдяки легкій для вивчення синтаксису та акценту на читабельність. Крім того, підтримка модулів і пакетів сприяє створенню модульних програм і повторному використанню коду. Python є мовою з відкритим кодом, тож численні незалежні програмісти постійно створюють для неї нові бібліотеки та функціональні можливості.

Векторна база даних з відкритим кодом Chroma [41], створена однойменним стартапом із Сан-Франциско, дозволяє створювати застосунки, зокрема для приблизного пошуку найближчих сусідів, пошуку зображень та систем рекомендацій в електронній комерції. Вона відома як легковагова векторна база даних, яку можна запускати на звичайному ноутбучі для швидкого прототипування, а також у публічних або приватних хмарних сервісах. Chroma використовує формат даних Apache Arrow для швидкого доступу до інформації [42].

Chroma можна запускати в режимі клієнт/сервер на одному вузлі та розгорнути її за допомогою Docker-контейнера або на орендованій машині в публічній хмарі. Також база доступна в хмарному сервісі Chroma Cloud – керуваному сервісі компанії з розгортанням на Amazon Web Services, Google Cloud Platform і Microsoft Azure.

Дана база даних, орієнтована на розробників, підтримує моделі векторизації від OpenAI, Google, Cohere, Hugging Face та інших. Вона має інтеграцію з Python та JavaScript у таких фреймворках, як LangChain, LlamaIndex та Braintrust, а також інтеграцію з Python-інструментами штучного інтелекту, зокрема Streamlit. Chroma надає клієнтські бібліотеки для різних мов програмування, включно з Python, JavaScript, Ruby, Java, Go, C#, Elixir та Rust.

Отже, в даному пункті наведено та описано засоби реалізації інтелектуальної системи автоматизованого підбору релевантних зображень.

### **3.2 Особливості використання спеціалізованих програмних компонентів**

У ході навчання та використання нейромережевої моделі для автоматизованого підбору релевантних зображень було використано дві основні бібліотеки. Цими бібліотеками є NumPy та Pillow. Також, для реалізації графічного інтерфейсу інтелектуальної системи було використано бібліотеку tkInter.

Бібліотека зображень Python (розширення PIL) – це пакет обробки зображень для мови Python. Він містить легкі інструменти обробки зображень, які допомагають редагувати, створювати та зберігати зображення. Підтримку бібліотеки зображень Python було припинено в 2011 році, але проєкт під назвою pillow розділив оригінальний проєкт PIL і додав до нього підтримку актуальних версій Python. Pillow була анонсована як заміна PIL для майбутнього використання. Pillow підтримує велику кількість форматів файлів зображень, включаючи BMP, PNG, JPEG і TIFF [43].

Сфери застосування pillow включають:

- завантаження та декодування зображень;
- обробка зображень;
- фільтрування зображень;
- покращення та перетворення зображень.

NumPy – це основний пакет для наукових обчислень на Python. Це бібліотека Python, яка надає об'єкти багатовимірного масиву, різноманітні похідні об'єкти (такі як замасковані масиви та матриці), а також набір процедур для швидких операцій над масивами, включаючи: математичні, логічні, маніпуляції формою, сортування, вибір, введення/виведення, дискретне перетворення Фур'є, основну лінійну алгебру, базові статистичні операції, випадкове моделювання та багато іншого [44].

У центрі пакета NumPy знаходиться об'єкт `ndarray`. Він інкапсулює  $n$ -вимірні масиви з однорідними типами даних, причому багато операцій виконуються у скомпільованому коді для забезпечення високої продуктивності.

Існує кілька важливих відмінностей між масивами NumPy та стандартними послідовностями Python. По-перше, масиви NumPy мають фіксований розмір при створенні, на відміну від списків Python, які можуть динамічно змінювати свій розмір. Якщо потрібно змінити розмір об'єкта `ndarray`, створюється новий масив, а початковий видаляється.

Крім того, всі елементи в масиві NumPy повинні мати один і той самий тип даних, отже, вони матимуть однаковий розмір у пам'яті. Винятком є масиви, які містять об'єкти Python (включаючи об'єкти NumPy), що дозволяє створювати масиви з елементами різного розміру.

Масиви NumPy значно полегшують виконання складних математичних і різноманітних операцій над великими обсягами даних. Зазвичай такі операції виконуються ефективніше і вимагають менше коду, ніж при використанні вбудованих послідовностей Python.

Tkinter – це стандартна бібліотека Python, яка надає набір інструментів та віджетів для створення настільних програм з графічним інтерфейсом. Tkinter входить до більшості інсталяцій Python, що робить її легкодоступною для розробників, які хочуть створювати GUI-додатки без додаткових інсталяцій чи бібліотек [45].

Спектр застосування Tkinter охоплює створення інтерактивних елементів інтерфейсу та комплексних програмних рішень. Це включає формування вікон та діалогових вікон для відображення інформації, отримання даних від користувача або представлення опцій. Tkinter також використовується для розробки повноцінних GUI для настільних програм, інтегруючи кнопки, меню та інші елементи взаємодії.

Крім того, Tkinter надає можливість інтеграції графічного інтерфейсу в існуючі консольні програми, що реалізує зручніше використання та спрощує

введення параметрів. Бібліотека містить набір вбудованих, а також підтримує створення власних віджетів для реалізації специфічних функціональних вимог.

Отже, було описано ключові бібліотеки, які були використані для створення інтелектуальної системи автоматизованого підбору релевантних зображень на основі семантичного пошуку. Зокрема, для числових обчислень, роботи з масивами та обробки даних, застосовувалася така бібліотека, як NumPy. Бібліотека Pillow використовувалася для роботи безпосередньо з зображеннями. Графічний інтерфейс системи, що забезпечує взаємодію користувача з функціоналом семантичного пошуку, була реалізована за допомогою стандартної бібліотеки Tkinter.

### 3.3 Структура інтелектуальної системи

Розуміння взаємодії між модулями застосунку дозволяє приймати кращі рішення при розробці та підтримці розробленого програмного забезпечення. Воно допомагає виявити потенційні слабкі місця розробленої інтелектуальної системи, оптимізувати її процеси та уникнути помилок.

Схема модульної структури інтелектуальної системи – це структурна схема, що відображає взаємозв'язки між компонентами системи та їхню взаємодію. Дана схема важлива для побудови, масштабування або модифікації будь-якої складної системи [46].

Модульна структура складається з 7 модулів, кожен з яких має різні функції. На рисунку 3.1 подано схему взаємозв'язків модулів .

Модуль `app.py` є стартовим файлом. Його завдання – запустити застосунок. Для цього він викликає `create_window()` із `logic.py`, що відкриває головне меню.

Модуль `logic.py` це ключовий вузол застосунку. Тут зібрано функції, які керують логікою звертання до інтерфейсу та запитів до БД. Саме, через модуль логіки, користувач може викликати функції вибору зображення та пошуку результатів, пошуку за текстом, запуск головного вікна, не маючи змоги при

цьому, взаємодіяти з цими функціями напряму, що забезпечує дотримання принципу інкапсуляції даних.

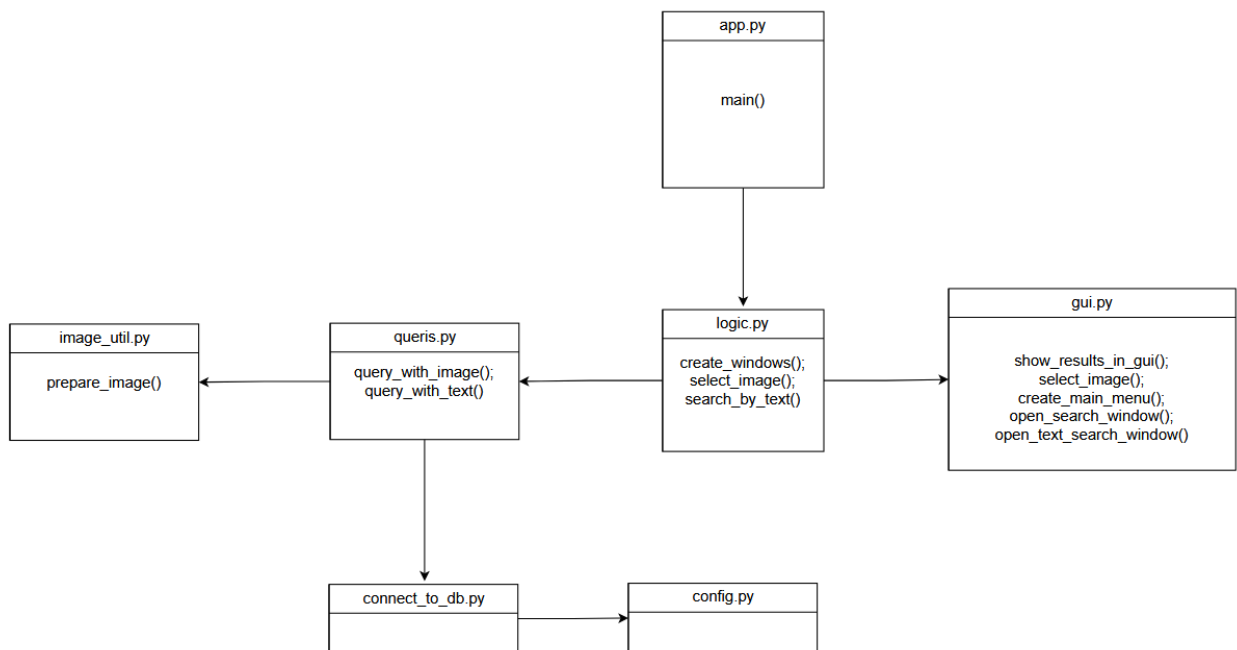


Рисунок 3.1 – Модульна структура інтелектуальної системи

Модуль `gui.py` відповідає за графічний інтерфейс. Тут створюється головне меню з кнопками, вікна для пошуку за текстом або зображенням. Цей файл взаємодіє з `logic.py`, щоб отримати результати пошуку та відобразити їх користувачу у зрозумілому вигляді.

У `queries.py` міститься основна логіка пошуку. В цьому модулі реалізовані функції, які беруть на вхід текст або зображення, обчислюють їхні вектори, і потім шукають схожі вектори у базі ChromaDB. В результаті повертається список найбільш схожих зображень разом із додатковою інформацією: назвою класу та відстанню (тобто, наскільки вони схожі).

Модуль `config.py` містить базові налаштування застосунку – шлях до бази та назву колекції. Це дозволяє централізовано керувати основними параметрами без необхідності дублювати їх у кожному модулі.

Модуль `connect_to_db.py` відповідає за зв'язок з векторною базою даних ChromaDB та ініціалізацію моделі векторизації даних OpenCLIP. У ньому створюється клієнт бази даних і завантажується модель для обчислення векторів.

Отже, було описано взаємодії між модулями інтелектуальної системи, та їх структуру. Чітке розмежування функцій між модулями дозволяє забезпечити інкапсуляцію, спростити модифікацію окремих компонентів та уникнути помилок під час внесення змін. Архітектура застосунку, яка включає окремі модулі для логіки, інтерфейсу, пошуку, конфігурації та взаємодії з базою даних, забезпечує структуровану та узгоджену взаємодію всіх частин системи.

### 3.4 Особливості реалізації інтелектуальної системи

Робота інтелектуальної системи підбору релевантних зображень ґрунтується на використанні векторної бази даних Chromadb для ефективного зберігання та пошуку зображень за їхнім семантичним змістом. Досягнення цієї функціональності забезпечується інтеграцією мультимодальної функції вбудовування CLIP, яка є ключовою для перетворення як візуальних даних, так і текстових запитів на уніфіковані векторні представлення. Це дає змогу системі знаходити подібні зображення, незалежно від того, чи запит надається у вигляді зображення, чи тексту. Взаємодія з цією базою даних, підготовка даних та зрозумілий графічний інтерфейс користувача є основними особливостями реалізації системи.

Однією з основних є функція, відповідальна за взаємодію з векторною базою даних, забезпечує її ініціалізацію та доступ до колекцій даних. Системне налаштування включає постійне з'єднання з базою даних за вказаним шляхом та отримання доступу до колекції, де зберігаються зображення. У цьому процесі задіяний завантажувач зображень та мультимодальна функція вбудовування. Нижче, даний процес описано псевдокодом:

```
ініціалізація_клієнта_chromadb(шлях_до_БД)
завантаження_зображень = ImageLoader()
функція_вбудовування_мультимодальна = OpenCLIPEmbeddingFunction()

мультимодальна_БД = отримати_колекцію_з_chromadb(
    ім'я_колекції,
    функція_вбудовування=функція_вбудовування_мультимодальна,
    завантажувач_даних=завантаження_зображень
)
```

Перед пошуком в базі даних запити проходять етап підготовки, який перетворює їх на формат, придатний для подальшої обробки. Наприклад, у випадку запиту за зображенням цей процес включає відкриття зображення, конвертацію його у формат RGB та перетворення на масив NumPy. Нижче, можна побачити псевдокод даного процесу:

```
функція підготувати_зображення(шлях_до_зображення):
    зображення_PIL = відкрити_зображення(шлях_до_зображення).конвертувати_в_RGB()
    повернути_масив_numpy(зображення_PIL)
```

Система підтримує два основні типи запитів, що дозволяють здійснювати пошук за різними вхідними даними. При пошуку за зображенням користувач надає зображення, яке трансформується у векторний ембедінг, після чого база даних знаходить семантично подібні зображення. У випадку пошуку за текстом користувач вводить текстовий запит, який також перетворюється на відповідний векторний ембедінг для подальшого порівняння з векторними представленнями зображень у базі даних.

```
функція запит_за_текстом(текст_запиту):
    якщо текст_запиту існує:
        результати = мультимодальна_БД.запит(
            запит_текстів=[текст_запиту],
            кількість_результатів=9,
            включити=['документи', 'відстані', 'метадані', 'дані']
        )
    повернути результати
```

Графічний інтерфейс користувача забезпечує інтуїтивно зрозумілу взаємодію із системою. Він дозволяє користувачеві обирати зображення або вводити текстові запити, а також ефективно відображає результати пошуку.

Головне меню (рисунок 3.2) надає користувачеві вибір між пошуком за зображенням та пошуком за текстом:

```
функція створити_головне_меню():
    root = створити_вікно_Tkinter()
    встановити_заголовок(root, "Головне меню")
    додати_кнопку(root, "Пошук за зображенням", відкрити_вікно_пошуку_за_зображенням)
```

```

додати_кнопку(root, "Пошук за текстом", відкрити_вікно_пошуку_за_текстом)
додати_кнопку(root, "Вихід", закрити_вікно)
запустити_головний_цикл(root)

```

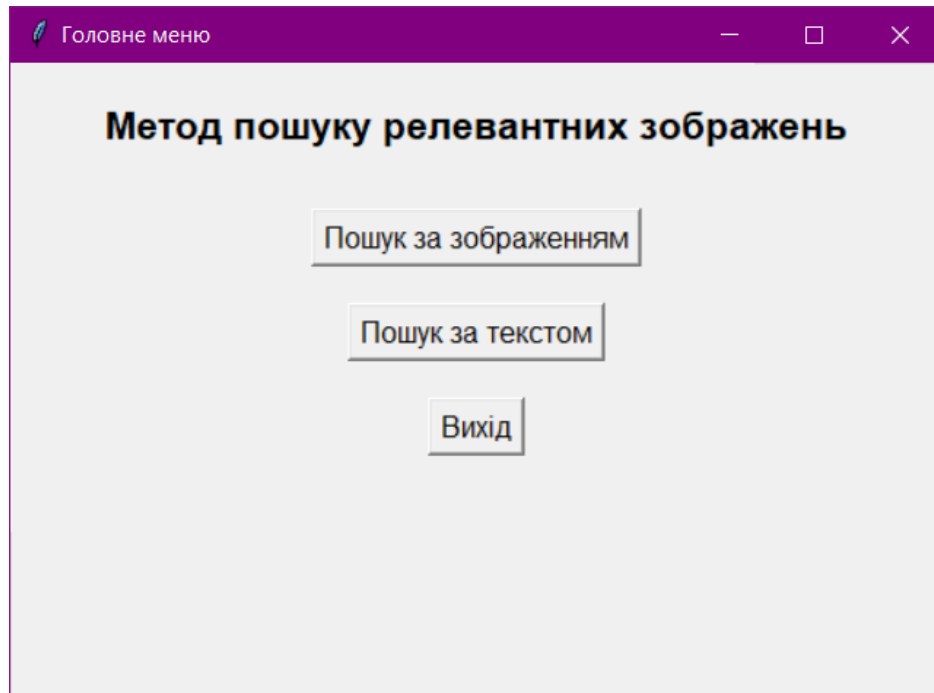


Рисунок 3.2 – Головне меню

Вікно пошуку за зображенням дозволяє користувачеві завантажувати зображення та ініціювати пошук. Результати відображаються у вигляді сітки з мініатюрами зображень, їхніми назвами класів та відстанню подібності:

```

функція відкрити_вікно_пошуку_за_зображенням(головне_вікно):
    нове_вікно = створити_нове_вікно(головне_вікно, "Пошук релевантних зображень")
    додати_мітку(нове_вікно, "Ваше зображення:")
    мітка_зображення_користувача = додати_мітку_для_зображення(нове_вікно)
    додати_кнопку(нове_вікно, "Обрати зображення", функція:
обробити_вибір_зображення(мітка_зображення_користувача, рамка_результатів))
    рамка_результатів = створити_прокручувану_рамку_для_результатів(нове_вікно)
    додати_кнопку(нове_вікно, "В меню", повернутись_назад)

```

Вікно пошуку за текстом (рисунок 3.3) містить поле для введення текстового запиту та відображає результати пошуку в аналогічному форматі:

```

функція відкрити_вікно_пошуку_за_текстом(головне_вікно):
    нове_вікно = створити_нове_вікно(головне_вікно, "Пошук релевантних зображень")
    додати_мітку(нове_вікно, "Введіть свій запит:")
    поле_введення_запиту = створити_поле_введення(нове_вікно)
    рамка_результатів = створити_прокручувану_рамку_для_результатів(нове_вікно)

```

```

додати_кнопку(нове_вікно, "Знайти", функція:
обробити_пошук_за_текстом(поле_введення_запиту.отримати_текст(), рамка_результатів))
додати_кнопку(нове_вікно, "В меню", повернутись_назад)

```

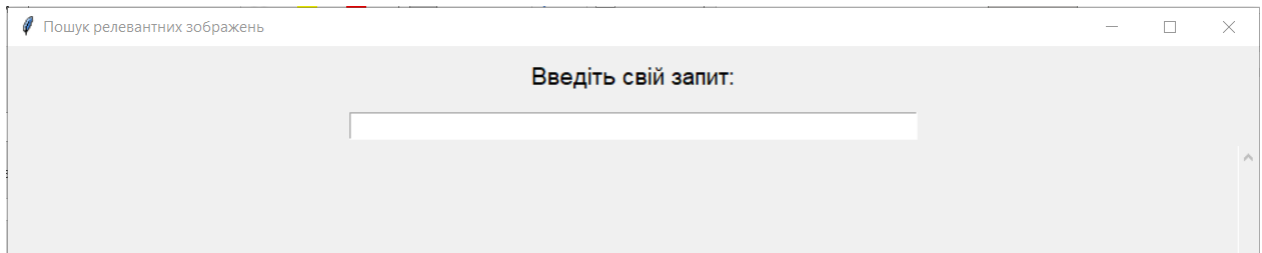


Рисунок 3.3 – Вікно пошуку за текстом

Функція `show_results_in_gui` відповідає за динамічне відображення отриманих результатів запиту в графічному інтерфейсі, створюючи мініатюри зображень, відображаючи їхні метадані та відстані:

```

функція показати_результати_в_gui(результати_запиту, рамка_результатів):
    очистити_рамку(рамка_результатів)
    для кожного результату в результатах_запиту:
        дані_зображення = отримати_дані_зображення_з_результату(результат)
        відстань = отримати_відстань_з_результату(результат)
        метадані = отримати_метадані_з_результату(результат)

        зображення =
конвертувати_дані_в_зображення_PIL(дані_зображення).змінити_розмір((250, 250))
        tk_зображення = конвертувати_зображення_PIL_в_Tkinter_Photo(зображення)
        зберегти_посилання_на_tk_зображення(tk_зображення)

        рамка_окремого_результату = створити_рамку_в_сітці(рамка_результатів)
        додати_мітку_із_зображенням(рамка_окремого_результату, tk_зображення)
        додати_мітку_з_текстом(рамка_окремого_результату, "{метадані['class_name']}
({відстань:.2f})")

```

Окрема функція виступає посередником між графічним інтерфейсом користувача та функціоналом запитів до бази даних, забезпечуючи коректну передачу даних та виклик відповідних функцій:

```

глобальний_список_посилань_на_зображення = []

функція вибрати_зображення(мітка_зображення_користувача, рамка_результатів):
    шлях_до_файлу = gui.вибрати_зображення(мітка_зображення_користувача)
    якщо шлях_до_файлу існує:
        результати_запиту = queries.запит_за_зображенням(шлях_до_файлу)
        gui.показати_результати_в_gui(результати_запиту, рамка_результатів)

функція пошук_за_текстом(текст_запиту, рамка_результатів):
    результати = queries.запит_за_текстом(текст_запиту)

```

```
gui.показати_результати_в_gui(результати, рамка_результатів)
```

```
функція створити_головне_вікно_додатку():
    gui.створити_головне_меню()
```

Отже, дана архітектура та функціонал забезпечує модульність та дозволяє легко розширювати можливості розробленої інтелектуальної системи, додавати нові методи пошуку або інтегрувати інші компоненти. Завдяки такому підходу, інтелектуальна система демонструє ефективний пошук інформації в мультимодальних даних, поєднуючи сучасні методи векторного пошуку зі зручним графічним інтерфейсом користувача.

### 3.5 Експериментальне дослідження запропонованого методу

У даному пункті проведено експериментальне дослідження запропонованого методу. Для цього досліджено донавчену модель векторизації CLIP досліджено за запропонованими у пункті 2.6 метриками на трьох різних етапах навчання моделі. Зокрема, дослідження статистичних метрик проводилося на кожній епосі моделі, протягом її навчання до 20-ї епохи.

На рисунку 3.4 можна побачити, як змінювалось середнє значення втрат, в ході донавчання моделі.

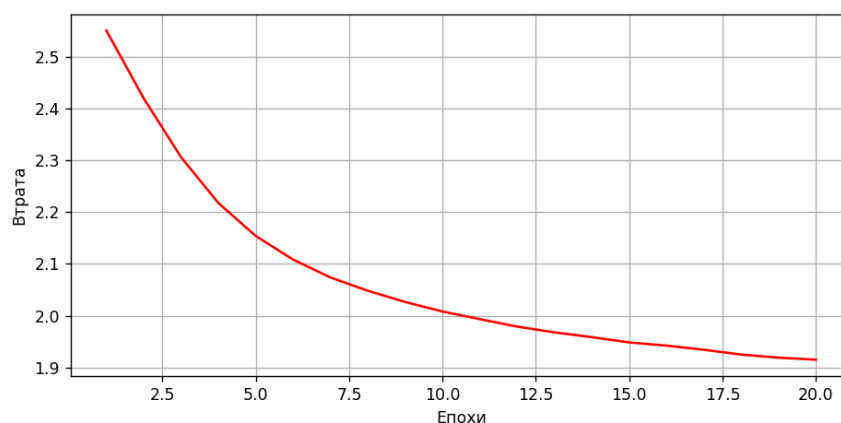


Рисунок 3.4 – Графік зміни середнього значення функції втрат

Середня функція втрат стабільно зменшувалася з 2.50 на 1-й епосі до 1.90 на 20-й епосі. Це значне зниження свідчить про те, що модель успішно

навчається розрізняти правильні пари зображення-текст від неправильних, мінімізуючи свої помилки. Постійне падіння втрат є індикатором того, що процес навчання ефективний, і модель стає все більш точною у своїх прогнозах.

На рисунку 3.5 зображено зміну значення метрики accuracy, в ході донавчання моделі.

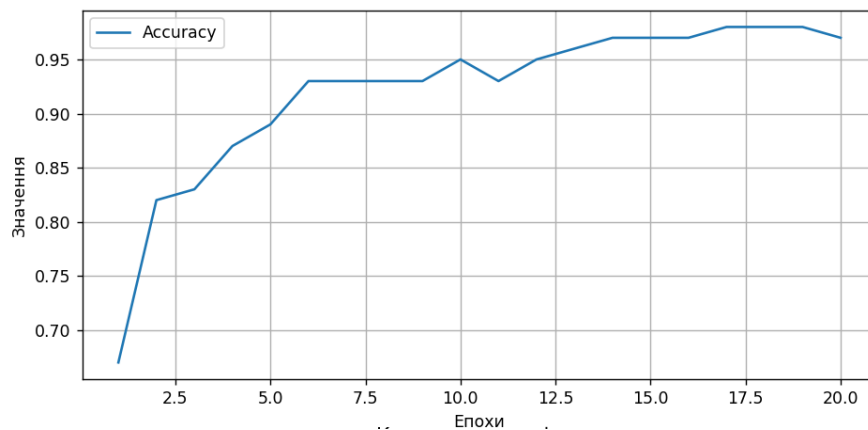


Рисунок 3.5 – Графік зміни значення accuracy (top-1)

Показник accuracy збільшився з 0.65 на 1-й епосі до 0.9700 на 20-й епосі. Це зростання свідчить про значне поліпшення здатності моделі до коректного зіставлення зображень із відповідними текстовими описами. Досягнення такого значення accuracy підтверджує високу ефективність класифікації моделі.

На рисунку 3.6 показаний графік зміни значення косинусної схожості, протягом донавчання моделі.

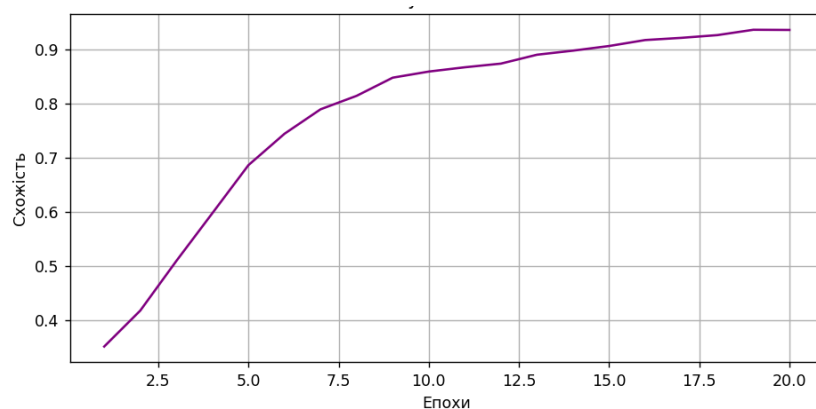


Рисунок 3.6 – Графік зміни значення косинусної схожості

Косинусна схожість збільшилася з 0.35 на 1-й епосі до 0.93 на 20-й епосі. Ця динаміка демонструє підвищення узгодженості між векторними представленнями зображень та їх відповідних текстових описів у просторі вбудовувань. Наближення цього значення до 1.0 вказує на успішне навчання моделі й на її здатність формувати семантично близькі репрезентації для відповідних мультимодальних об'єктів.

Для перевірки того, наскільки точно модель виконує поставлену перед нею задачу пошуку, було проведене дослідження, яке полягало в аналізі отриманих результатів експертами. Було використано текстовий пошук, з наступним запитом – «night sky» (нічне небо). Як результат, було отримано 9 зображень, які інтелектуальна система вважала найбільш релевантними, для заданого запиту. На рисунку 3.7 зображено пошуковий запит та отримані результати.

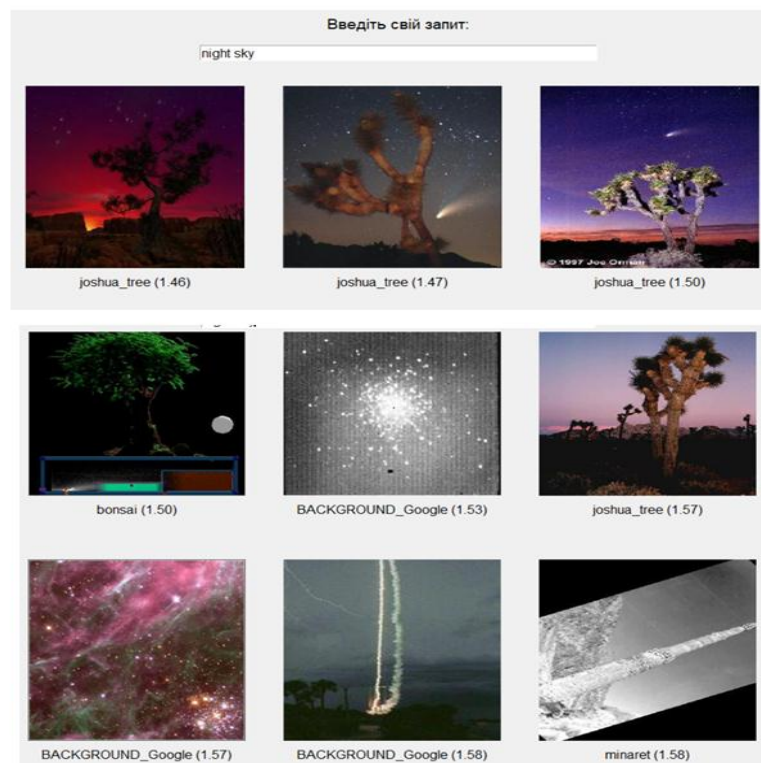


Рисунок 3.7 – Результат текстового пошукового запиту

Експерти оцінювали подібність зображення за системою, описаною в таблиці 3.1.

Таблиця 3.1 – Шкала оцінки експертів

Оцінка	Значення релевантності
0	Релеванте
1	Частково релевантне
2	Нерелевантне

Такий підхід дозволяє співставити та порівняти оцінку релевантності, яку дають результату експерти та векторну відстань до запиту, яку повертає, разом з результатом, інтелектуальна система.

В таблиці 3.2 наведено експертні рішення щодо здійсненого наведеного на рисунку 3.7 пошукового запиту.

Таблиця 3.2 – Класифікація зображень експертами

Зображення	Експерт №1	Експерт №2	Експерт №3	Середня оцінка експертів
joshua_tree (1.46)	0	0	0	0
joshua_tree (1.50)	0	0	0	0
joshua_tree (1.47)	0	0	0	0
joshua_tree (1.57)	1	0	2	1
BACKGROUND_Google (1.53)	1	2	2	1.66
BACKGROUND_Google (1.57)	1	2	2	1.66
BACKGROUND_Google (1.58)	1	2	1	1.33
bonsai (1.50)	2	2	2	2
minaret (1.58)	2	2	2	2
<b>Загальна середня оцінка результату</b>				<b>1.07</b>

Коли інтелектуальна система повертає результат, його елементами є: зображення, клас та косинусна відстань до вектору запиту (значення від 0 до 2, де 0 це ідентичний вектор, а 2 це «протилежний» вектор). На основі цих даних можна робити висновки про точність результатів.

Як можна побачити з отриманих результатів, косинусна відстань даних зображень знаходиться в діапазоні від 1.46 до 1.58. Це говорить про те, що система не вважає результати справді релевантними, і більше класифікує їх як частково релевантні. Такий результат є очевидним, оскільки в базі даних немає зображень виключно зоряного неба, лише такі, де воно є другорядним об'єктом на фоні.

Оцінки експертів відзначились більшою категоричністю. Так, наприклад, перші три зображення були оцінені, як релевантні, оскільки, хоч і на фоні, але нічне небо чітко помітне, а *bonsai* (1.50) та *minaret* (1.58) було визначено, як нерелевантні, оскільки не містили нічного неба, як такого. Втім, загальна середня оцінка експертів про даний набір результатів, все виявилась наближеною до того, як його оцінила система.

Отже, в результаті експериментального дослідження було обраховано та проаналізовано показники метрик, які вказують на те, що навчання моделі CLIP відбувалось стабільно. Дослідження експертами встановило, що запропонована реалізація наведеного в роботі методу дозволяє покращити процес автоматизованого підбору релевантних зображень.

### **3.6 Висновки до розділу 3**

У даному розділі було описано засоби розробки та структуру інтелектуальної системи. Реалізований у інтелектуальній системі метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних забезпечує векторизацію та пошук

релевантних зображень за текстовим або візуальним запитом. Також, було проведене дослідження запропонованого методу.

Для програмної реалізації інтелектуальної системи було обрано Python як основну мову програмування, а Visual Studio Code використано як середовище розробки. Ключовим елементом для зберігання та ефективного пошуку векторних представлень даних стала ChromaDB – векторна база даних, оптимізована для задач семантичного пошуку. Застосовані технології забезпечують масштабованість та стабільну роботу системи.

Функціонування системи забезпечується інтеграцією спеціалізованих бібліотек – Pillow та NumPy. Pillow застосовується для попередньої обробки зображень, включаючи їх завантаження та декодування. NumPy використовується для високоефективних операцій над багатовимірними масивами, що допомагає виконувати обробку та порівняння векторних даних. Для розробки графічного інтерфейсу системи було використано стандартну бібліотеку Tkinter. Це дозволило створити зручний візуальний інтерфейс для управління пошуком та відображенням результатів.

Розроблена інтелектуальна система має модульну структуру, що складається з семи взаємопов'язаних модулів. Архітектура передбачає логічне розмежування функцій, що включають запуск застосунку, керування логікою, створення графічного інтерфейсу користувача, реалізацію основних алгоритмів пошуку, зберігання конфігураційних налаштувань та забезпечення підключення до векторної бази даних і моделі векторизації. Така організація сприяє інкапсуляції, спрощенню модифікації та зменшенню ризику помилок.

Експериментальне дослідження показало наступні результати: середнє значення функції втрат зменшувалося протягом навчання, а значення асигасу та косинусної схожості зросли, що вказує на підвищення точності та повноти класифікації та про точніше узгодження між векторними представленнями зображень та їхніх описів. Експертна оцінка результатів текстового пошуку підтвердила здатність інтелектуальної системи надавати релевантні результати, навіть за умови обмеженого наповнення бази даних.

## Загальні висновки

Метою кваліфікаційної роботи було покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних. У кваліфікаційній роботі було запропоновано відповідний метод та реалізовано інтелектуальну систему автоматизованого підбору релевантних зображень, що дозволило досягнути поставленої мети.

Для досягнення мети виконано такі задачі:

- виконано дослідження предметної області для задачі пошуку релевантних зображень;
- розроблено метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних;
- виконано програмну реалізацію створеного методу автоматизованого підбору релевантних зображень на основі семантичного пошуку;
- проведено дослідження роботи методу з використанням розробленої інтелектуальної системи.

Функціонально інтелектуальна система забезпечує:

- векторизацію вхідних текстових запитів та зображень;
- обчислення схожості векторів у багатовимірному просторі;
- підбір релевантних зображень на основі змістової відповідності;
- інтерактивну взаємодію з користувачем через програмний інтерфейс.

Покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку досягається шляхом підвищення відповідності результатів пошуковому запиту через використання донавченої моделі CLIP у поєднанні з векторними базами даних.

Тому розроблений метод автоматизованого підбору релевантних зображень на основі семантичного пошуку має потенціал для практичного застосування у сферах, де потрібен пошук візуального контенту – зокрема, в

електронній комерції, цифрових архівах, медіабібліотеках та системах рекомендацій.

Подальше вдосконалення автоматизованого підбору релевантних зображень базується на розширенні її функціональних можливостей та оптимізації ключових аспектів. Одним із пріоритетних напрямків є розширення мовної підтримки. Також важливим кроком у вдосконаленні є використання різноманітнішого набору даних для донавчання моделі. Збільшення обсягу та варіативності доступного візуального контенту дозволить розширити область застосування системи та підвищити точність пошуку шляхом збільшення кількості релевантних результатів, що можуть бути знайдені.

## Перелік посилань

1. What is visual search and how does it work? *iRonin.IT*. URL: <https://www.ironin.it/blog/what-is-visual-search.html> (date of access: 04.06.2025).
2. A comprehensive guide to feature extraction in python. *viso.ai*. URL: <https://viso.ai/deep-learning/feature-extraction-in-python/> (date of access: 04.06.2025).
3. The computer vision pipeline, part 4: feature extraction. *Medium*. URL: <https://manningbooks.medium.com/the-computer-vision-pipeline-part-4-feature-extraction-6343ef063588> (date of access: 04.06.2025).
4. Image meta search. *Wikipedia, the free encyclopedia*. URL: [https://en.wikipedia.org/wiki/Image\\_meta\\_search](https://en.wikipedia.org/wiki/Image_meta_search) (date of access: 04.06.2025).
5. What is semantic search, and how does it work?. *Google Cloud*. URL: <https://cloud.google.com/discover/what-is-semantic-search> (date of access: 04.06.2025).
6. Короткий посібник із розуміння алгоритму KNN. *Unite.AI*. URL: <https://www.unite.ai/uk/a-quick-guide-to-knn-algorithm/> (дата звернення: 04.06.2025).
7. Що таке KNN (K-найближчі сусіди)? *Unite.AI*. URL: <https://www.unite.ai/uk/чому-дорівнює-k-найближчих-сусідів/> (дата звернення: 04.06.2025).
8. Your ultimate guide to vector search. *Oracle*. URL: <https://www.oracle.com/ua/database/vector-search/> (date of access: 18.06.2025).
9. What is a vector database?. *CLOUDFLARE*. URL: <https://www.cloudflare.com/learning/ai/what-is-vector-database/>.
10. What is content-based image retrieval? *Baeldung on Computer Science*. URL: <https://www.baeldung.com/cs/cbir-tbir> (date of access: 04.06.2025).
11. What is computer vision? *IBM - United States*. URL: <https://www.ibm.com/think/topics/computer-vision> (date of access: 04.06.2025).

12. Working flow of a CBIR system. *ResearchGate*. URL: [https://www.researchgate.net/figure/Working-flow-of-a-CBIR-system\\_fig3\\_281200105](https://www.researchgate.net/figure/Working-flow-of-a-CBIR-system_fig3_281200105) (date of access: 04.06.2025).
13. Dagan A., Guy I., Novgorodov S. Shop by image: characterizing visual search in e-commerce. *Information retrieval journal*. 2023. Vol. 26, no. 1. URL: <https://doi.org/10.1007/s10791-023-09418-1> (date of access: 06.06.2025).
14. Salih S. F., Abdulla A. A. An effective bi-layer content-based image retrieval technique. *The journal of supercomputing*. 2022. URL: <https://doi.org/10.1007/s11227-022-04748-1> (date of access: 06.06.2025).
15. Late E., Ruotsalainen H., Kumpulainen S. Image searching in an open photograph archive: search tactics and faced barriers in historical research. *International journal on digital libraries*. 2024. URL: <https://doi.org/10.1007/s00799-023-00390-1> (date of access: 06.06.2025).
16. Google reverse image search: everything you need to know. *SmartFrame*. URL: <https://smartframe.io/blog/google-reverse-image-search-everything-you-need-to-know/> (date of access: 04.06.2025).
17. How to do a reverse image search from your phone. *PCMag*. URL: <https://www.pcmag.com/how-to/how-to-do-a-reverse-image-search-from-your-phone> (date of access: 04.06.2025).
18. Ultimate guide for finding products by picture on alibaba. *Alibaba.com Seller Central*. URL: <https://seller.alibaba.com/businessblogs/ultimate-guide-for-finding-products-by-picture-on-alibaba-px002caqa> (date of access: 04.06.2025).
19. Finding similar images with alibaba cloud image search. *Alibaba Cloud Community*. URL: [https://www.alibabacloud.com/blog/finding-similar-images-with-alibaba-cloud-image-search\\_593805](https://www.alibabacloud.com/blog/finding-similar-images-with-alibaba-cloud-image-search_593805) (date of access: 04.06.2025).
20. Finding similar images with alibaba cloud image search. *Medium*. URL: <https://alibaba-cloud.medium.com/finding-similar-images-with-alibaba-cloud-image-search-7eb525d458a5> (date of access: 04.06.2025).
21. Діаграма послідовності (sequence diagrams). *Maxym Zosym*. URL: <https://www.maxzosim.com/sequence-diagrams/> (дата звернення: 04.06.2025).

22. What is activity diagram?. *Visual Paradigm*. URL: <https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-activity-diagram/> (date of access: 04.06.2025).
23. CLIP: connecting text and images. *OpenAI*. URL: <https://openai.com/index/clip/> (date of access: 04.06.2025).
24. CLIP: Contrastive Language-Image Pre-Training. *viso.ai*. URL: <https://viso.ai/deep-learning/clip-machine-learning/> (date of access: 04.06.2025).
25. Contrastive Language-Image Pre-training. *Wikipedia, the free encyclopedia*. URL: [https://en.wikipedia.org/wiki/Contrastive\\_Language-Image\\_Pre-training](https://en.wikipedia.org/wiki/Contrastive_Language-Image_Pre-training) (date of access: 04.06.2025).
26. BLIP: bootstrapping language-image pre-training for unified vision-language understanding and generation. *Salesforce*. URL: <https://www.salesforce.com/blog/blip-bootstrapping-language-image-pretraining/> (date of access: 04.06.2025).
27. Understanding BLIP : a huggingface model. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/understanding-blip-a-huggingface-model/> (date of access: 04.06.2025).
28. BLIP explained. *Papers With Code*. URL: <https://paperswithcode.com/method/blip> (date of access: 04.06.2025).
29. Datasets definition. *Encord*. URL: <https://encord.com/glossary/datasets-definition/> (date of access: 04.06.2025).
30. Caltech 101. *CaltechDATA*. URL: <https://data.caltech.edu/records/mzrjq-6wc02> (date of access: 04.06.2025).
31. Caltech-101 dataset. *Kaggle*. URL: <https://www.kaggle.com/datasets/imbikramsaha/caltech-101> (date of access: 04.06.2025).
32. Caltech101. *TensorFlow*. URL: <https://www.tensorflow.org/datasets/catalog/caltech101> (date of access: 04.06.2025).
33. Object classification with caltech 101. *Encord*. URL: <https://encord.com/blog/object-classification-caltech-101/> (date of access: 04.06.2025).

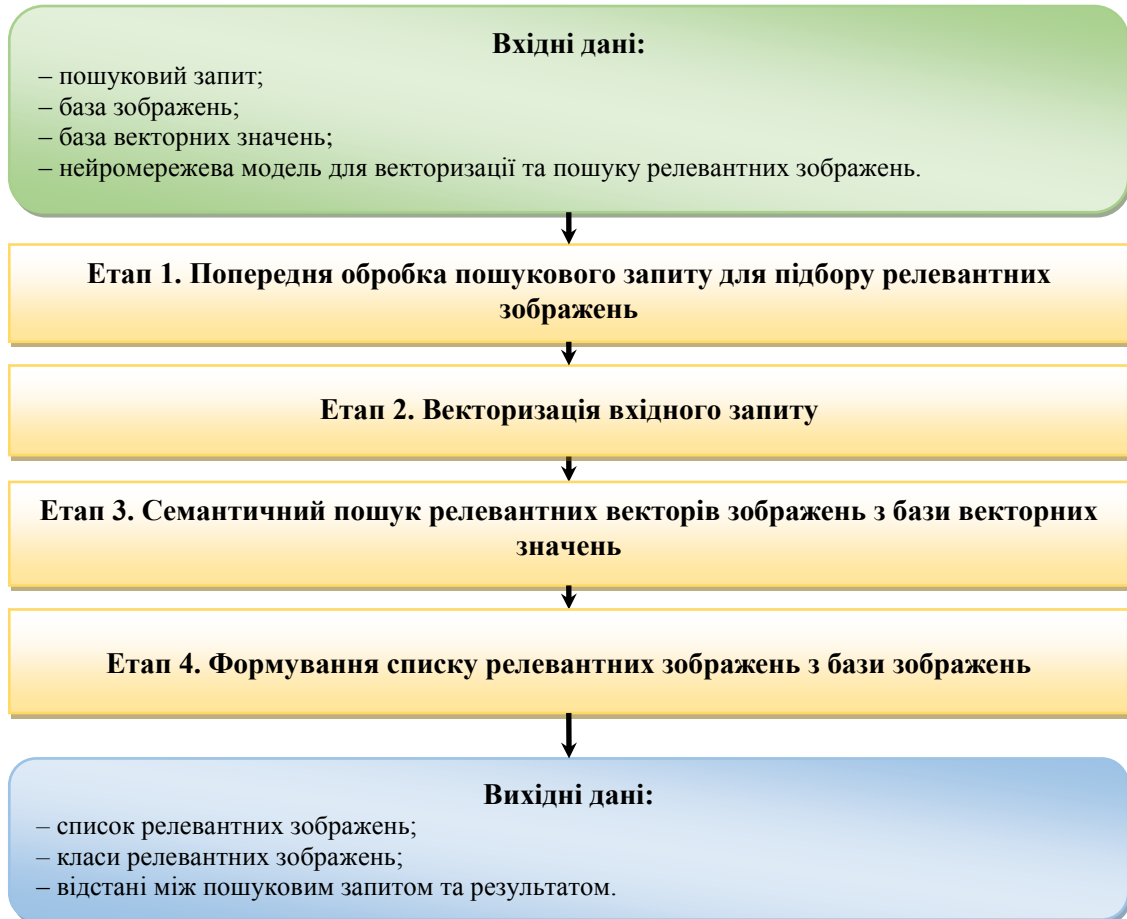
34. Average loss. *CloudFactory Computer Vision Wiki*. URL: <https://wiki.cloudfactory.com/docs/mp-wiki/loss/average-loss> (date of access: 04.06.2025).
35. Top-1 error rate. *Giskard*. URL: <https://www.giskard.ai/glossary/top-1-error-rate> (date of access: 12.06.2025).
36. Cosine similarity. *ScienceDirect*. URL: <https://www.sciencedirect.com/topics/computer-science/cosine-similarity> (date of access: 04.06.2025).
37. Visual studio code - code editing. redefined. *Visual Studio Code*. URL: <https://code.visualstudio.com/> (date of access: 04.06.2025).
38. What is visual studio code? Microsoft's extensible code editor. *InfoWorld*. URL: <https://www.infoworld.com/article/2335960/what-is-visual-studio-code-microsofts-extensible-code-editor.html> (date of access: 04.06.2025).
39. Welcome to python.org. *Python.org*. URL: <https://www.python.org/> (date of access: 04.06.2025).
40. What is python programming language?. *Teradata*. URL: <https://www.teradata.com/insights/data-platform/what-is-python-programming-language> (date of access: 04.06.2025).
41. Chroma. *Chroma*. URL: <https://www.trychroma.com/> (date of access: 04.06.2025).
42. How oracle AI vector search stacks up against chroma for similarity search. *Oracle.com*. URL: <https://www.oracle.com/ua/database/vector-database/chromadb/> (date of access: 04.06.2025).
43. Python: Pillow (a fork of PIL). *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/python-pillow-a-fork-of-pil/> (date of access: 04.06.2025).
44. What is numpy? – numpy v2.2 manual. *NumPy*. URL: <https://numpy.org/doc/stable/user/whatisnumpy.html> (date of access: 04.06.2025).
45. What is tkinter for python?. *GeeksforGeeks*. URL: <https://www.geeksforgeeks.org/introduction-to-tkinter/> (date of access: 10.06.2025).

46. Why is application architecture so important?. *BairesDev*. URL: <https://www.bairesdev.com/blog/why-is-app-architecture-so-important/> (date of access: 04.06.2025).

# ДОДАТКИ

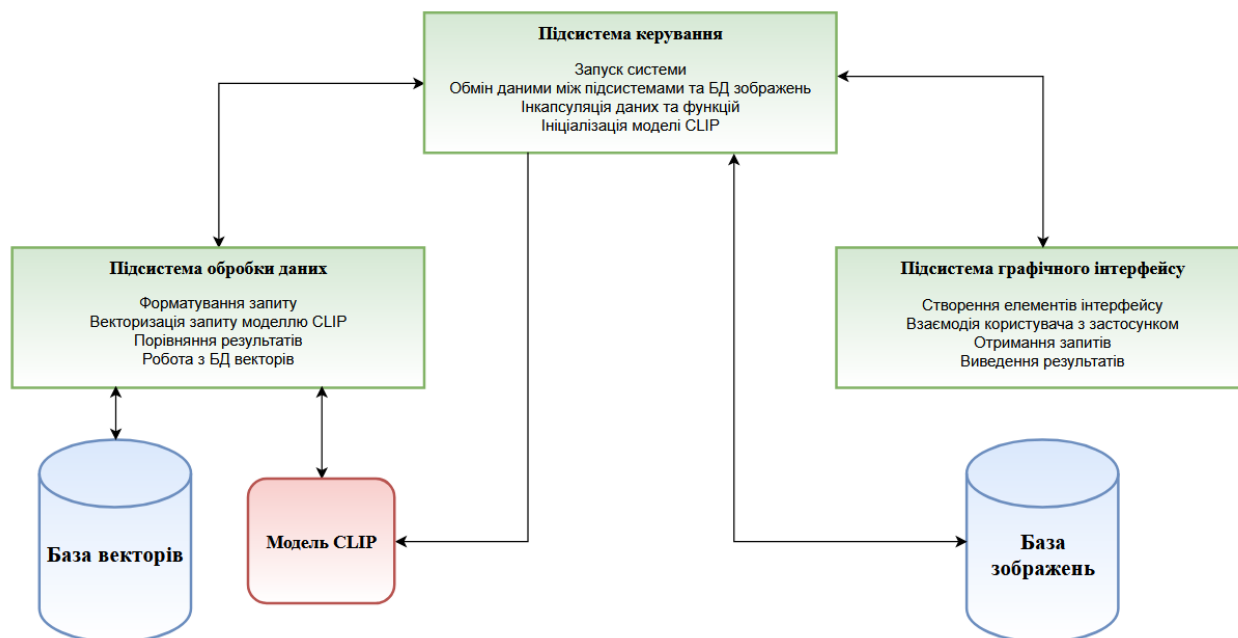
## Додаток А

### Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних



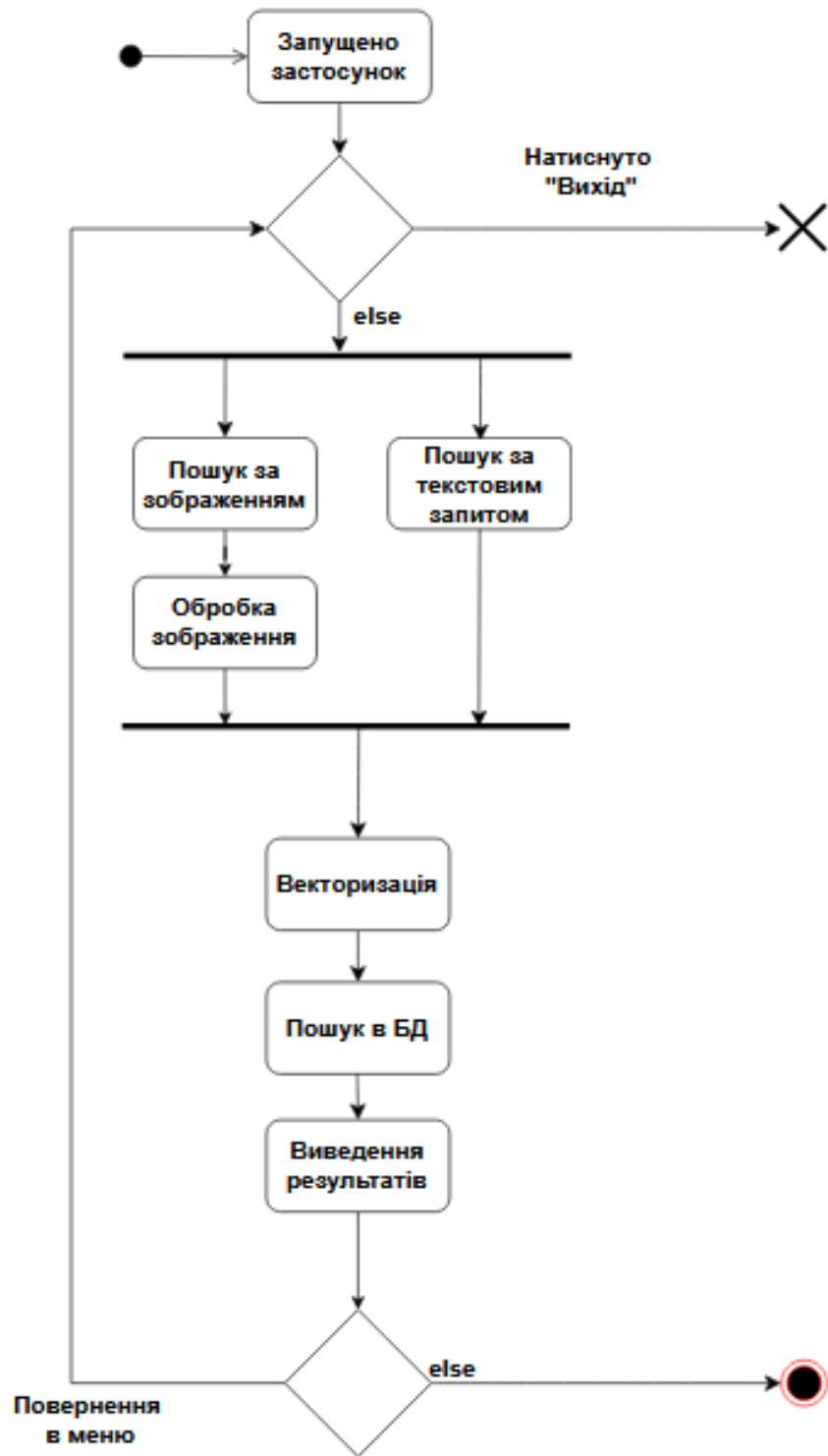
## Додаток Б

### Проектна архітектура



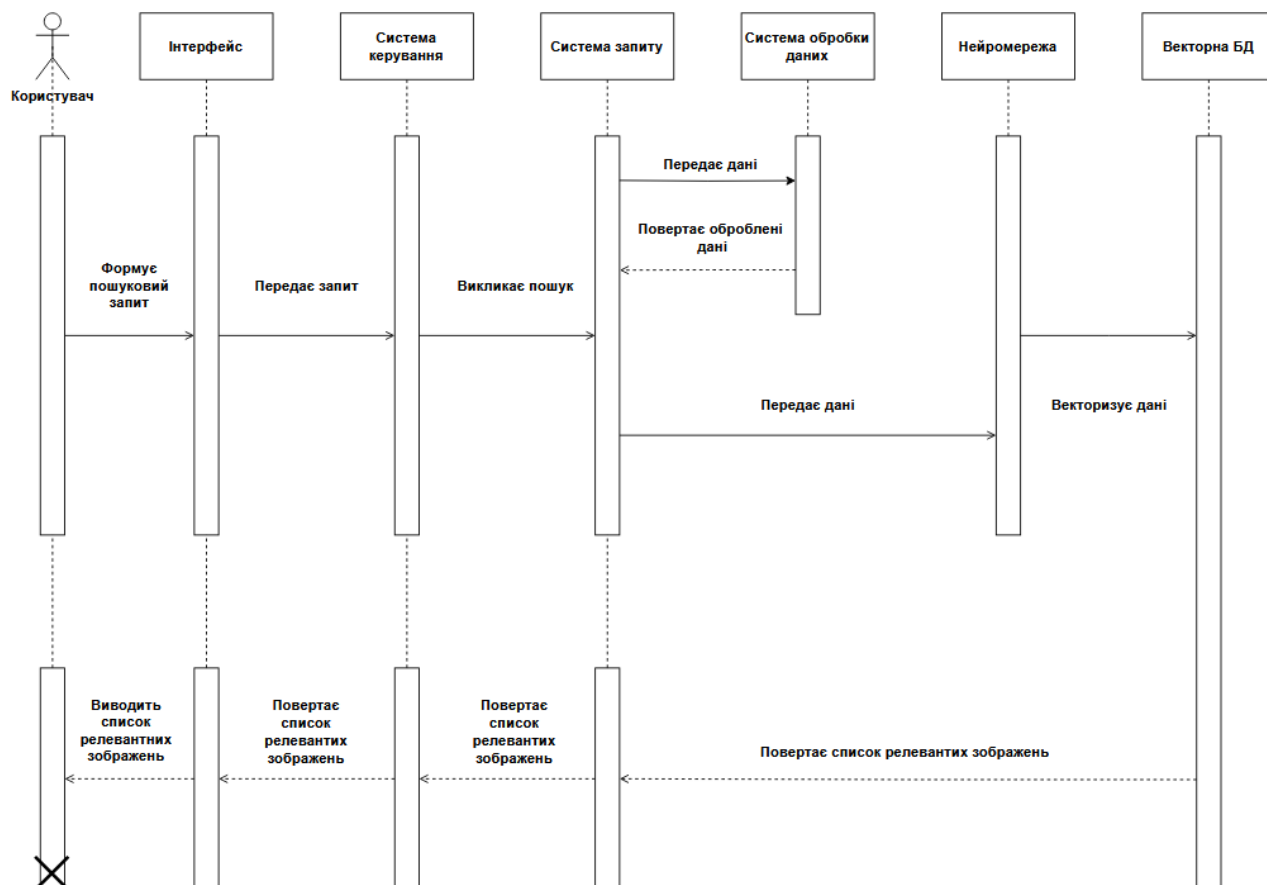
### Додаток В

### Діаграма активності



## Додаток Г

### Діаграма послідовності



## Додаток Д

### Програмні коди

Вихідний код, використаний у дослідженні, доступний у репозиторії GitHub: <https://github.com/Razonenaz/SemanticSearchOfRelevantImages> (дата звернення: 06.06.2025).

File	Commit Message	Time
README.md	Update README.md	4 days ago
app.py	Add files via upload	4 days ago
c_db.py	Add files via upload	4 days ago
config.py	Add files via upload	4 days ago
connect_to_db.py	Add files via upload	4 days ago
gui.py	Add files via upload	4 days ago
image_util.py	Add files via upload	4 days ago
logic.py	Add files via upload	4 days ago
queries.py	Add files via upload	4 days ago

Рисунок Д.1 – Головна сторінка репозиторію

Структура репозиторію наступна:

- модуль запуску застосунку (app.py). Є стартовим файлом проекту. Його завдання – ініціалізувати застосунок, викликаючи функцію create\_window() із модуля логіки, що відкриває головне вікно програми;

- модуль логіки застосунку (logic.py). Відповідає за взаємодію між інтерфейсом та функціями пошуку. Забезпечує реалізацію основних функцій: вибору зображень, пошуку за текстом або зображенням, запуску головного вікна. Реалізований з дотриманням принципу інкапсуляції – доступ до функціоналу здійснюється через інтерфейс, без прямої взаємодії з логікою;

- модуль графічного інтерфейсу (gui.py). Формує головне меню, вікна пошуку за текстом або зображенням, а також виводить результати. Взаємодіє з

logic.py для виконання дій, ініційованих користувачем, та забезпечує зручний візуальний відгук;

– модуль пошукових запитів (queries.py). Містить основну логіку пошуку: обчислює векторні подання зображень або тексту, виконує пошук у базі ChromaDB та повертає результати з назвами класів і метрикою подібності;

– модуль конфігурацій (config.py). Зберігає основні параметри застосунку, такі як шлях до бази даних і назва колекції, дозволяючи централізовано змінювати налаштування без потреби правити кожен модуль окремо;

– модуль підключення до бази та моделі (connect\_to\_db.py). Відповідає за ініціалізацію клієнта бази даних ChromaDB і завантаження моделі OpenCLIP для векторизації. Також здійснює підключення до колекції, яка використовується для пошуку релевантних зображень.

## Додаток Е

### Презентація

КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА

# МЕТОД АВТОМАТИЗОВАНОГО ПІДБОРУ РЕЛЕВАНТНИХ ЗОБРАЖЕНЬ НА ОСНОВІ СЕМАНТИЧНОГО ПОШУКУ ЗАСОБАМИ ВЕКТОРНИХ БАЗ ДАНИХ



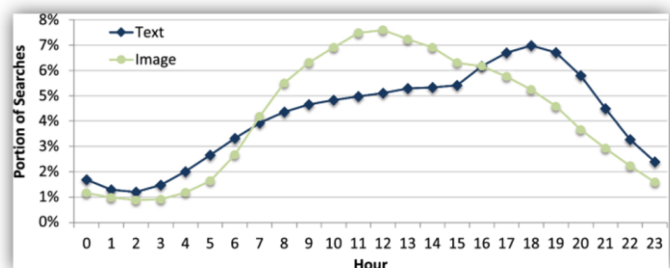
**Виконав:**  
*студент групи КН-21-2*  
**Назар Остапченко**



**Керівник:**  
*асистент каф. КН*  
**Олена ТИЩЕНКО**

## Актуальність

Пошук зображень є фундаментальним елементом сучасного світу, та важливою складовою інтернету, що забезпечує швидкий та інтуїтивно зрозумілий доступ до великого простору візуального вмісту. Завдяки прогресу штучного інтелекту та технологій машинного навчання пошук зображень продовжує розвиватися, стаючи дедалі складнішим і здатним задовольняти потреби користувачів усе точнішими та надійнішими способами.



Графік текстових пошукових запитів та запитів за зображенням, погодинно, протягом дня

## Мета і задачі роботи

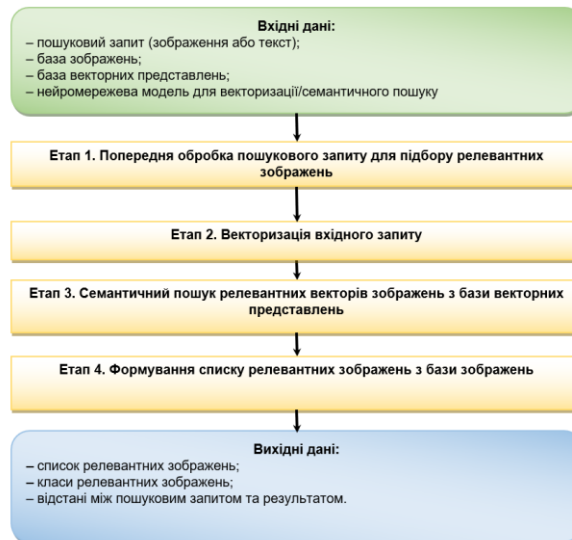
**Об'єкт дослідження** – процес автоматизованого підбору релевантних зображень на основі семантичного пошуку.

**Предмет дослідження** – методи та технології семантичного пошуку для автоматизованого підбору релевантних зображень.

**Метою кваліфікаційної роботи бакалавра** є підвищення якості процесу пошуку релевантних зображень шляхом використання семантичного пошуку та векторних баз даних.

Для виконання завдання даної роботи необхідно розробити метод автоматизованого підбору релевантних зображень на основі семантичного пошуку з використанням векторних баз даних. Провести аналіз сучасних підходів до семантичного пошуку, зокрема методів векторного представлення даних і алгоритмів обробки візуального контенту. Створити архітектуру інформаційної системи для генерації векторів, зберігання даних і швидкого пошуку відповідних зображень. Реалізувати та протестувати інтелектуальну систему, оцінити ефективність методу за показниками точності семантичного пошуку.

## Схема методу підбору релевантних зображень



Модель CLIP (Contrastive Language–Image Pretraining) – це універсальний засіб векторизації, який поєднує можливості нейронних мереж для зображень та тексту, дозволяючи здійснювати ефективне зіставлення зображень і текстових описів у спільному векторному просторі.

BLIP (Bootstrapping Language-Image Pre-training) – це модель, об'єднує сфери обробки природної мови та комп'ютерного зору. Завдяки масштабному попередньому навчанню на мільйонах пар «зображення–текст», BLIP ефективно виконує задачу автоматичного створення підписів до зображень, крос-модальний пошук

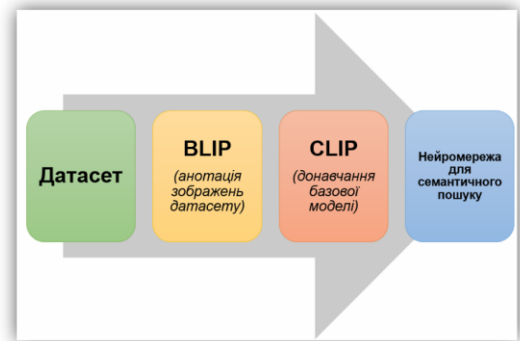


Схема донавчання моделі векторизації даних

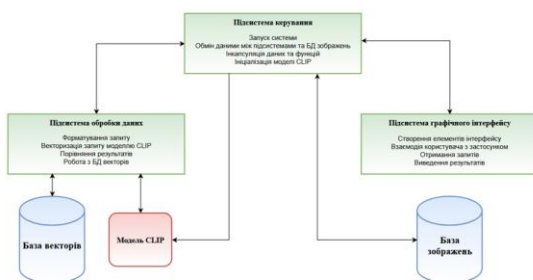


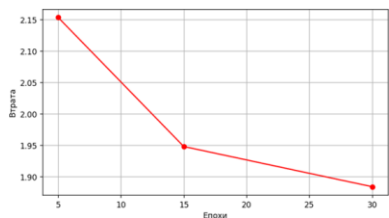
Схема підсистем



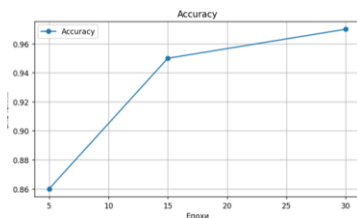
Схема програмних модулів



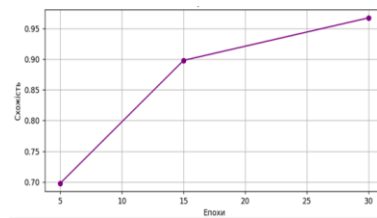
## Статистичні метрики



Графік зміни середнього значення функції втрат



Графік зміни значення accuracy (top-1)



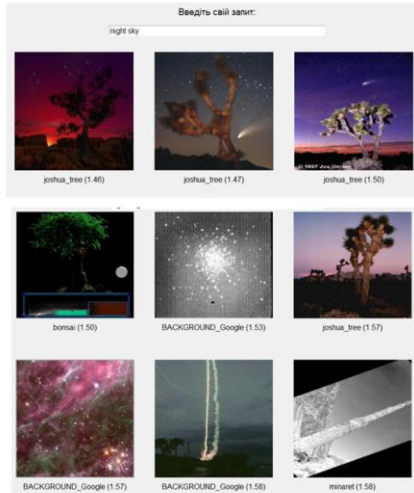
Графік зміни значення косинусної схожості

## Експериментальне дослідження релевантності результатів

Оцінка	Значення релевантності
0	Релевантне
1	Частково релевантне
2	Нерелевантне

Шкала оцінки отриманого результату експертами

## Експериментальне дослідження релевантності результатів



Результат за запитом «nigh sky»

Зображення	Експерт №1	Експерт №2	Експерт №3	Середня оцінка експертів
joshua_tree (1.46)	0	0	0	0
joshua_tree (1.50)	0	0	0	0
joshua_tree (1.47)	0	0	0	0
joshua_tree (1.57)	1	0	2	1
BACKGROUND_Google (1.53)	1	2	2	1.66
BACKGROUND_Google (1.57)	1	2	2	1.66
BACKGROUND_Google (1.58)	1	2	1	1.33
bonsai (1.50)	2	2	2	2
minaret (1.58)	2	2	2	2
<b>Загальна середня оцінка результату</b>				<b>1.07</b>

## Висновки

Отже, в результаті виконання КРБ було створено інтелектуальну систему, яка дозволяє виконувати підбір релевантних зображень з використанням семантичного пошуку та векторних баз даних. Було проведено теоретичний аналіз сучасних методів семантичного пошуку та обґрунтовано вибір підходу для розробки власного методу. Результати дослідження показали високу ефективність запропонованого методу за якісними та кількісними показниками.

## Anti-Plagiarism (UA) v-15.281 Educational

The maximum coincidence with one document **2.0%**

Dictionaries check: en\_US, ru\_RU, ua\_UA. **Errors in the documents: 9%**

ID: 246901 Title: КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних Added in a DB: 2025-06-19 Authors: Назар ОСТАПЧЕНКО Heads: Олена ТИЩЕНКО Consultants: Opponents:	Document		Sum coincidence on the DB	
	Symbols	Lexemes	Symbols	Lexemes
	72743	1054	3302 (5%)	51 (5%)

### Plagiarism sources

ID	Description	Plagiarism presence in the document	
		Symbols	Lexemes

## Протокол аналізу звіту подібності науковим керівником

Заявляю, що я ознайомився (-лась) з Повним звітом подібності, який був згенерований Системою виявлення і запобігання плагіату щодо роботи:

**Автор:** Назар ОСТАПЧЕНКО

**Співавтор:**

**Назва:** КВАЛІФІКАЦІЙНА РОБОТА БАКАЛАВРА на тему Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

**Науковий керівник:** Олена ТИЩЕНКО, асистент. каф. КН

**Підрозділ:** Кафедра комп'ютерних наук

**Коефіцієнт подібності 1:** 9.3%

**Коефіцієнт подібності 2:** 3.8%

**Мікропробіли:** 0

**Заміна букв:** 1

**Інтервали:** 0

**Білі знаки:** 50

**Дата створення звіту:** 2025-06-19 13:45:42.0

Після аналізу Звіту подібності констатую наступне:

Запозичення, виявлені в роботі є законними і не є плагіатом. Рівень подібності не перевищує допустимої межі. Таким чином робота незалежна і приймається.

Запозичення не є плагіатом, але перевищено граничне значення рівня подібностей. Таким чином робота повертається на доопрацювання.

Виявлено запозичення і плагіат або навмисні текстові спотворення (маніпуляції), як передбачувані спроби укриття плагіату, які роблять роботу невідповідною вимогам законодавства (Ст. 32. ЗУ Про вищу освіту, пункт 3.1, Ст. 42. ЗУ Про освіту) та вимог НАЗЯВО (Критерій 5), а також кодексу етики і процедурам. Таким чином робота не приймається.

Обґрунтування:

2025-06-19

Дата

експерт

*Dr. Petrovich P.P.*

# РІШЕННЯ ЕКСПЕРТНОЇ КОМІСІЇ КАФЕДРИ КОМП'ЮТЕРНИХ НАУК

## ПРО ДОПУСК КВАЛІФІКАЦІЙНОЇ РОБОТИ ДО ЗАХИСТУ

Назва кваліфікаційної роботи Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

Автор студент групи КН-21-2 Назар Остапченко

Освітня програма Комп'ютерні науки

Рівень вищої освіти перший (бакалаврський)

Спеціальність 122 – Комп'ютерні науки

Науковий керівник: асистент каф. комп'ютерних наук Олена Тищенко

На основі аналізу кваліфікаційної роботи на дотримання вимог академічної доброчесності (у т.ч. відсутності ознак академічного плагіату) з урахуванням результатів перевірки роботи спеціалізованим програмними засобами комісія зробила такий висновок:

№	Висновок	Позначка про відповідність
1	Ознаки академічного плагіату	
1.1	Запозичення, виявлені в роботі, є законними і не є академічним плагіатом (далі – зазначаються підстави віднесення запозичень до правомірних, якщо потрібно). Робота приймається до захисту.	<i>відповідає</i>
1.2	Виявлені запозичення не є академічним плагіатом, розміщені в розділах, які не описують безпосередньо авторське дослідження, але кількість цитат перевищує обсяг, виправданий поставленою метою роботи (далі – зазначаються детальні та аргументовані підстави віднесення запозичень до правомірних). Робота приймається до захисту, але має бути відкоригована.	
1.3	Виявлені запозичення не є академічним плагіатом, але частково розміщені в розділах, які описують безпосередньо авторське дослідження, а кількість цитат перевищує обсяг, виправданий поставленою метою роботи. Робота може бути допущена до захисту після того як буде відкоригована та доопрацьована і успішно пройде повторну перевірку на академічний плагіат.	
1.4	Робота містить навмисні текстові спотворення, передбачувані спроби укриття текстових запозичень або інші прояви академічного плагіату. Робота містить фабрикацію або фальсифікацію даних. Робота не допускається до захисту.	
2	Інші види порушень академічної доброчесності	<i>відсутні</i>

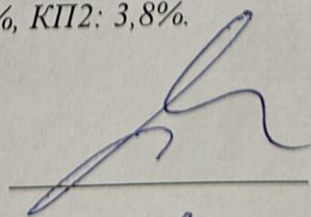
### Підтвердження:

*Запозичення, виявлені в роботі Назара Остапченка, не є плагіатом, оскільки: запозичення розміщені в розділі огляду існуючих підходів, не описують безпосередньо авторську роботу і не стосуються її результатів; усі запозичення фрагментарні; до запозичень входять фрагменти, які не мають авторства і містять поширені конструкції та загальновідомі терміни, скорочення. Рівень подібності не перевищує допустимої межі. Таким чином, робота є законною та приймається до захисту.*

*Обсяг запозичень, визначений системами виявлення збігів/ідентичності/схожості:*  
 - за системою Anti-Plagiarism: 2%;  
 - за системою StrikePlagiarism КП1: 9,3%, КП2: 3,8%.

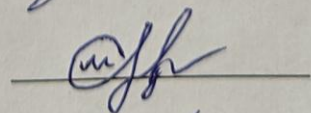
19.06.2025

Завідувач кафедри



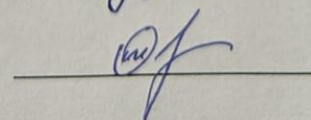
Олександр БАРМАК

Гарант освітньої програми



Олександр МАЗУРЕЦЬ

Керівник кваліфікаційної роботи



Олена ТИЩЕНКО



**ВІДГУК НАУКОВОГО КЕРІВНИКА  
на кваліфікаційну роботу бакалавра**

студента гр. КН-21-2 Остапченка Назара Вячеславовича

за темою Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

**1. Актуальність теми**

Тема автоматизованого підбору релевантних зображень на основі семантичного пошуку є надзвичайно актуальною в умовах стрімкого зростання обсягів мультимедійних даних та необхідності ефективного їх опрацювання. Використання векторних баз даних і моделей, що забезпечують семантичне представлення зображень, дозволяє значно підвищити точність та швидкість пошуку, що має важливе практичне значення для різних галузей, зокрема для медіа, електронної комерції та наукових досліджень. Реалізація методу, який базується на сучасних підходах до векторизації даних, відповідає сучасним вимогам інформаційних технологій та сприяє підвищенню якості користувацького досвіду, що підтверджує актуальність і значущість виконаної кваліфікаційної роботи.

**2. Відповідність роботи предметній області Стандарту спеціальності 122 Комп'ютерні науки**

За стандартом, а саме описом предметної області, об'єктом роботи є процес автоматизованого підбору релевантних зображень на основі семантичного пошуку. Метою роботи є покращення процесу автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних. При вирішенні поставленої задачі використано методи та технології семантичного пошуку для автоматизованого підбору релевантних зображень. Тому результати виконання кваліфікаційної роботи бакалавра відповідають стандарту бакалавра спеціальності 122 – Комп'ютерні науки.

**3. Професійні та особистісні якості бакалавра**

У процесі підготовки бакалаврської роботи студент продемонстрував низку сформованих професійно значущих якостей: уважність до змісту завдань, вміння аргументовано відстоювати прийняті методологічні рішення, відповідальне ставлення до термінів виконання та наукової якості результатів.

#### **4. Ступінь самостійності під час виконання кваліфікаційної роботи**

Отримані результати кваліфікаційної роботи свідчать про повну самостійність студента у виконанні поставлених завдань.

#### **5. Ступінь оволодіння методами дослідження**

Студент продемонстрував належний рівень оволодіння сучасними методами дослідження, обґрунтовано застосовуючи їх у процесі виконання кваліфікаційної роботи. Використані методи відповідають завданням дослідження та забезпечують досягнення поставленої мети, що свідчить про сформовані практичні навички наукової діяльності та методологічну підготовку здобувача.

#### **6. Повнота та якість розкриття теми роботи**

Тема кваліфікаційної роботи розкрита повно та всебічно, з урахуванням актуальних наукових підходів і практичних аспектів. Студент здійснив систематичний аналіз існуючих підходів, що забезпечило цілісне і глибоке розуміння предметної області. Якість викладу матеріалу відповідає вимогам академічної доброчесності та наукової строгості, що підвищує наукову цінність роботи.

#### **7. Логічність, послідовність, аргументованість, літературна грамотність викладення матеріалу**

Викладення матеріалу у кваліфікаційній роботі вирізняється логічністю та послідовністю, що забезпечує чітке розкриття теми дослідження. Аргументація наукових положень є обґрунтованою і підтверджується відповідними джерелами. Текст роботи відповідає вимогам літературної грамотності та академічного стилю, що сприяє зрозумілості та цілісності подання дослідження.

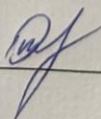
#### **8. Можливість практичного застосування кваліфікаційної роботи бакалавра, окремих її частин**

Розроблений у роботі метод автоматизованого підбору релевантних зображень на основі семантичного пошуку із застосуванням векторних баз даних може бути використаний у системах управління медіаконтентом, електронних бібліотеках та платформах електронної комерції для підвищення точності і швидкості пошуку зображень за змістом.

#### **9. Висновок про можливість допуску кваліфікаційної роботи бакалавра до захисту, на яку оцінку заслуговує робота**

Враховуючи високий рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно».

Керівник \_\_\_\_\_



асистент каф. КН Олена ТИЩЕНКО



## РЕЦЕНЗІЯ

### на кваліфікаційну роботу бакалавра

студента *гр. КН-21-2 Остапченка Назара Вячеславовича*

за темою: Метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних

#### 1. Актуальність обраної теми

Предмет дослідження, пов'язаний із автоматизацією підбору релевантних зображень через семантичний пошук із застосуванням векторних баз даних, є важливим з огляду на сучасні виклики у сфері обробки великих обсягів мультимедійної інформації. У контексті зростаючої потреби у швидкому та точному відборі візуального контенту, розробка ефективних алгоритмів семантичної індексації та пошуку має суттєве значення для підвищення продуктивності інформаційних систем. Враховуючи розвиток технологій штучного інтелекту та їх впровадження у практичні рішення, дана робота має вагомое теоретичне й практичне значення, що підкреслює її актуальність.

#### 2. Повнота розкриття мети та завдань роботи

У кваліфікаційній роботі мета та завдання сформульовані чітко й логічно, повною мірою відображають суть дослідження та забезпечують цілісність його реалізації. Їх послідовне розкриття в структурі роботи свідчить про усвідомлене оперування науковим апаратом і здатність здобувача до самостійного аналізу проблематику.

#### 3. Зміст кожного розділу роботи

Зміст кожного розділу кваліфікаційної роботи є послідовним, узгодженим із загальною логікою дослідження та спрямованим на реалізацію поставленої мети й завдань. Перший розділ присвячений аналізу інформаційних моделей в області автоматизованого підбору зображень за запитом. У другому розділі наведено розроблений метод автоматизованого підбору релевантних зображень на основі семантичного пошуку засобами векторних баз даних. У третьому розділі проведено експериментальне дослідження методу та висвітлено його результати.

4. Оцінка розробленої інформаційної системи, її практична цінність

Система дозволяє користувачам ефективно знаходити необхідний контент у великих масивах даних, що оптимізує процеси управління медіаресурсами. Практична цінність полягає у підвищенні релевантності пошукових результатів з урахуванням специфіки запитів.

5. Якість оформлення кваліфікаційної роботи бакалавра

Оформлення кваліфікаційної роботи відповідає встановленим вимогам та нормам академічного стилю та спеціальності комп'ютерних наук. Текст має чітку структуру, що полегшує сприйняття матеріалу, а використання ілюстрацій, таблиць і бібліографічних посилань виконано коректно та систематично.

6. Недоліки кваліфікаційної роботи бакалавра

У роботі присутнє використання неуніфікованих термінів для позначення одних і тих самих понять. Недостатньо точне формулювання висновків, особливо проміжних. Недостатня увага до опису обмежень дослідження. Деякі рисунки (2.4, 2.6) мають малу чіткість, варто було б зробити чіткішими. Зауважу, що виявлені недоліки є незначними й не впливають на загальну наукову цінність отриманих результатів.

7. Загальний висновок (допускається чи не допускається до захисту), та оцінка на яку заслуговує кваліфікаційна робота.

Враховуючи рівень виконання та забезпечення усіх необхідних вимог, робота може бути допущена до захисту. Рекомендована оцінка «відмінно».

Рецензент

Мартинович В.В.  
Завідувач кафедри  
АІТІТР

