Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра програмної інженерії
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня
Бакалавр

(назва освітнього ступеня)

на тему: **Розробка програмного забезпечення моделі генерації текстових описів зображень на основі Vision-Language підходів**

Виконав(ла): студент(ка) 4 курсу, групи СП-42
спеціальності 121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

	(підпис)	Мигаль З. Я. (прізвище та ініціали)
Керівник	(підпис)	Цебрій О.Р. (прізвище та ініціали)
Нормоконтроль	(підпис)	Стоянов Ю.М. (прізвище та ініціали)
Завідувач кафедри	(підпис)	Петрик М.Р. (прізвище та ініціали)
Рецензент	(підпис)	Луцик Н.С. (прізвище та ініціали)

Тернопіль
2026

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра програмної інженерії
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри

_____ _____
(підпис) (прізвище та ініціали)
« » 20___ р.

З А В Д А Н Н Я
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавра
(назва освітнього ступеня)

за спеціальністю 121 Інженерія програмного забезпечення
(шифр і назва спеціальності)

студенту Мигаль Зоряна Ярославівна
(прізвище, ім'я, по батькові)

1. Тема роботи Розробка програмного забезпечення моделі генерації текстових описів зображень на основі Vision-Language підходів

Керівник роботи Цебрій О.Р., канд. фіз.-мат. наук, доц.
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «___» _____ 20___ року № _____.

2. Термін подання студентом завершеної роботи

3. Вихідні дані до роботи Предметна область, завдання, вимоги та специфікація, програмне рішення, методичні вказівки

4. Зміст роботи (перелік питань, які потрібно розробити)
Вступна частина
Аналіз предметної області та теоретичних основ
Визначення методики реалізації моделі
Реалізація моделі
Визначення основних аспектів охорони праці та безпеки життєдіяльності
Висновки роботи

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
Слайди презентації та діаграми процесів

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності,	Мариненко С. Ю., доц.		
основи охорони праці			
Нормоконтроль	Стоянов Ю.М. к.т.н., доц. каф. ПІ		

7. Дата видачі завдання _____

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1	Отримання завдання		
2	Аналіз завдання		
3	Виконання розділу 1: Аналіз предметної області та сучасних Vision-Language підходів		
4	Виконання розділу 2: Проєктування та розробка програмної системи генерації описів зображень		
5	Виконання розділу 3: Тестування, оцінка ефективності та впровадження системи		
6	Оформлення пояснювальної записки		
7	Оформлення графічного та презентаційного матеріалу		
8	Попередній захист		
9	Захист		

Студент _____ Мигаль З. Я.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Цебрій О.Р.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Розробка моделі машинного навчання для генерації текстових описів зображень на основі Vision-Language підходів // Кваліфікаційна робота освітнього рівня «Бакалавр» // Мигаль Зоряна Ярославівна // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем і програмної інженерії, кафедра програмної інженерії, група СП-42 // Тернопіль, 2026 // С. ___, рис. – ___, табл. – ___, додат. – ___, бібліогр. – ___.

Ключові слова: машинне навчання; глибинне навчання; Vision-Language модель; генерація текстових описів зображень; image captioning; комп'ютерний зір; штучний інтелект; BLIP; Transformer; MS COCO.

Кваліфікаційна робота присвячена розробці моделі машинного навчання для генерації текстових описів зображень на основі Vision-Language підходів. Проведено аналіз сучасних моделей і наборів даних для задачі image captioning, реалізовано та протестовано модель генерації описів зображень.

У роботі розглянуто особливості поєднання методів комп'ютерного зору та обробки природної мови для створення мультимодальних систем. Також виконано оцінювання якості згенерованих описів із використанням сучасних метрик, що дозволило визначити ефективність запропонованого підходу.

Об'єкт дослідження — процес автоматичної генерації текстових описів зображень.

Предмет дослідження — методи та моделі Vision-Language для задачі image captioning.

Практичним результатом є програмна система, що автоматично формує текстові описи зображень і може застосовуватися в системах доступності, пошуку та аналізу візуального контенту.

ABSTRACT

Development of a Machine Learning Model for Image Caption Generation Based on Vision-Language Approaches // Bachelor's Qualification Thesis // Zoriana Myhal // Ternopil Ivan Puluj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Software Engineering, Group SP-42 // Ternopil, 2026 // P. ___, fig. – ___, tabl. – ___, annexes – ___, references – ___.

Keywords: machine learning; deep learning; Vision-Language model; image captioning; computer vision; artificial intelligence; BLIP; Transformer; MS COCO.

The qualification thesis is devoted to the development of a machine learning model for image caption generation based on Vision-Language approaches. An analysis of modern models and datasets for the image captioning task was conducted, and an image caption generation model was implemented and tested.

The thesis examines the integration of computer vision and natural language processing methods for building multimodal systems. The quality of generated captions was evaluated using modern evaluation metrics, which made it possible to assess the effectiveness of the proposed approach.

The object of research is the process of automatic image caption generation.

The subject of research is Vision-Language methods and models for the image captioning task.

The practical result of the thesis is a software system capable of automatically generating textual descriptions of images. The developed solution can be applied in accessibility systems, image retrieval systems, and visual content analysis applications.

ЗМІСТ

ПЕРЕЛІК СКОРОЧЕНЬ

ML – Machine Learning (машинне навчання)

AI – Artificial Intelligence (штучний інтелект)

DL – Deep Learning (глибинне навчання)

EDA – Exploratory Data Analysis (розвідувальний аналіз даних)

CSV – Comma-Separated Values (текстовий формат даних)

LR – Logistic Regression (логістична регресія)

DT – Decision Tree (дерево рішень)

kNN – k-Nearest Neighbors (метод k найближчих сусідів)

SVM – Support Vector Machine (метод опорних векторів)

NB – Naive Bayes (наївний баєсівський класифікатор)

NN – Neural Network (нейронна мережа)

RF – Random Forest (випадковий ліс)

GBM – Gradient Boosting Machine (градієнтний бустинг)

XGB – XGBoost (Extreme Gradient Boosting)

LGBM – LightGBM (Light Gradient Boosting Machine)

CB – CatBoost (Categorical Boosting)

SMOTE – Synthetic Minority Oversampling Technique (синтетичне збільшення вибірки меншості)

ROC – Receiver Operating Characteristic (крива робочих характеристик приймача)

AUC – Area Under Curve (площа під кривою)

API – Application Programming Interface (інтерфейс прикладного програмування)

CPU – Central Processing Unit (центральний процесор)

RAM – Random Access Memory (оперативна пам’ять)

JSON – JavaScript Object Notation (формат даних JSON)

ВСТУП

У сучасних інформаційних системах, де обсяги мультимедійного контенту постійно зростають, автоматичний аналіз зображень стає однією з ключових задач у сфері штучного інтелекту. Особливої актуальності набувають системи, здатні не лише розпізнавати окремі об’єкти на зображенні, але й формувати повноцінний текстовий опис сцени природною мовою. Окреме значення технології генерації описів зображень має для створення засобів доступності для людей із порушеннями зору, де автоматично сформований опис дозволяє отримати текстове представлення візуального контенту.

Задача генерації текстових описів зображень належить до мультимодальних задач штучного інтелекту, оскільки поєднує методи комп’ютерного зору та обробки природної мови. На відміну від класичних задач класифікації або детекції об’єктів, image captioning потребує не лише виділення окремих елементів сцени, але й розуміння їх взаємозв’язків, контексту та формування граматично правильного речення. Це значно ускладнює процес побудови моделей та підвищує вимоги до якості навчальних даних і архітектури нейронної мережі.

Традиційні підходи до генерації описів зображень переважно базувалися на поєднанні згорткових нейронних мереж для виділення візуальних ознак та рекурентних нейронних мереж для генерації тексту. Хоча такі моделі забезпечували базову якість генерації, вони мали низку обмежень, пов’язаних із недостатнім урахуванням глобального контексту зображення, складністю обробки довгих текстових залежностей та обмеженою узагальнювальною здатністю. Крім того, класичні CNN-RNN архітектури часто формують шаблонні або недостатньо інформативні описи, що негативно впливає на природність згенерованого тексту.

Подальший розвиток технологій штучного інтелекту привів до появи сучасних Vision-Language підходів, які базуються на Transformer-архітектурах та мультимодальному навчанні. Використання Vision-Language моделей, зокрема BLIP, CLIP, ViT та інших мультимодальних архітектур, забезпечує значне покращення якості генерації тексту порівняно з традиційними підходами.

Додатковою перевагою є можливість використання попередньо навчених моделей та масштабних наборів даних, таких як MS COCO Captions або Flickr8k/Flickr30k, що дозволяє підвищити точність роботи системи та скоротити витрати на навчання моделей.

Практична цінність систем генерації текстових описів зображень полягає у можливості автоматизації процесів аналізу та структуризації візуального контенту. Такі системи можуть застосовуватися у вебсервісах, пошукових системах, соціальних мережах, електронних бібліотеках та мультимедійних платформах для автоматичного створення описів зображень, тегування контенту та покращення пошуку за зображеннями.

Основною метою даної роботи є розробка та експериментальне дослідження моделі машинного навчання для автоматичної генерації текстових описів зображень на основі Vision-Language підходів. Особливу увагу приділено аналізу сучасних мультимодальних архітектур, вибору ефективної моделі для задачі image captioning та дослідженню якості генерації тексту залежно від використаних методів обробки даних і параметрів навчання.

Експериментальна частина роботи передбачає підготовку та обробку наборів даних MS COCO Captions і Flickr8k/Flickr30k, навчання моделі генерації описів зображень та оцінювання якості її роботи із використанням сучасних метрик, зокрема BLEU, ROUGE, METEOR та CIDEr. У процесі дослідження планується провести порівняння різних Vision-Language архітектур, оцінити вплив параметрів навчання на якість генерації та проаналізувати приклади сформованих текстових описів. Окрему увагу буде приділено аналізу здатності моделі формувати контекстно-залежні та граматично коректні речення.

Очікується, що використання сучасних Vision-Language моделей дозволить підвищити якість генерації текстових описів у порівнянні з класичними CNN-RNN підходами та забезпечить більш природне формування тексту. Результатом роботи має стати програмна система, здатна автоматично генерувати описи зображень англійською або українською мовою та забезпечувати можливість подальшої інтеграції у вебсервіси або мультимедійні платформи.

1. АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ТА СУЧАСНИХ VISION-LANGUAGE ПІДХОДІВ

Розвиток мультимодальних систем штучного інтелекту суттєво розширив можливості автоматичного аналізу візуальної інформації та генерації тексту природною мовою. Одним із напрямів таких систем є задача генерації текстових описів зображень, яка поєднує методи комп'ютерного зору та обробки природної мови. Для побудови ефективних моделей image captioning необхідним є аналіз сучасних Vision-Language підходів, архітектур нейронних мереж та методів обробки мультимодальних даних. Особливу увагу приділяють Transformer-моделям та мультимодальним архітектурам, які забезпечують кращу узгодженість між візуальними та текстовими представленнями інформації [1].

Важливим етапом дослідження є аналіз сучасних наборів даних для задачі генерації описів зображень, зокрема MS COCO Captions, Flickr8k та Flickr30k, що широко використовуються для навчання та тестування Vision-Language моделей. Крім цього, необхідним є дослідження програмних засобів і бібліотек для реалізації системи, вибір оптимальних технологій розробки та формування вимог до програмного забезпечення. Проведений аналіз дозволяє визначити найбільш ефективні підходи до побудови системи генерації текстових описів зображень та сформувати основу для подальшого проєктування і реалізації моделі машинного навчання.

1.1 Загальна характеристика задачі генерації текстових описів зображень

Задача image captioning полягає в автоматичному формуванні текстового опису на основі вхідного зображення. На відміну від класичної класифікації зображень, система повинна не лише визначити об'єкти сцени, але й встановити взаємозв'язки між ними та сформувати граматично правильне речення природною

мовою. Це робить задачу генерації описів однією з найбільш складних у сфері мультимодального штучного інтелекту [2].

Сучасні системи генерації описів зазвичай складаються з двох основних компонентів: візуального енкодера та текстового декодера. Візуальний енкодер виконує аналіз зображення та виділення ознак, тоді як декодер формує текстовий опис на основі отриманих візуальних представлень. Загальну схему роботи системи генерації текстових описів зображень наведено на рисунку 1.1.

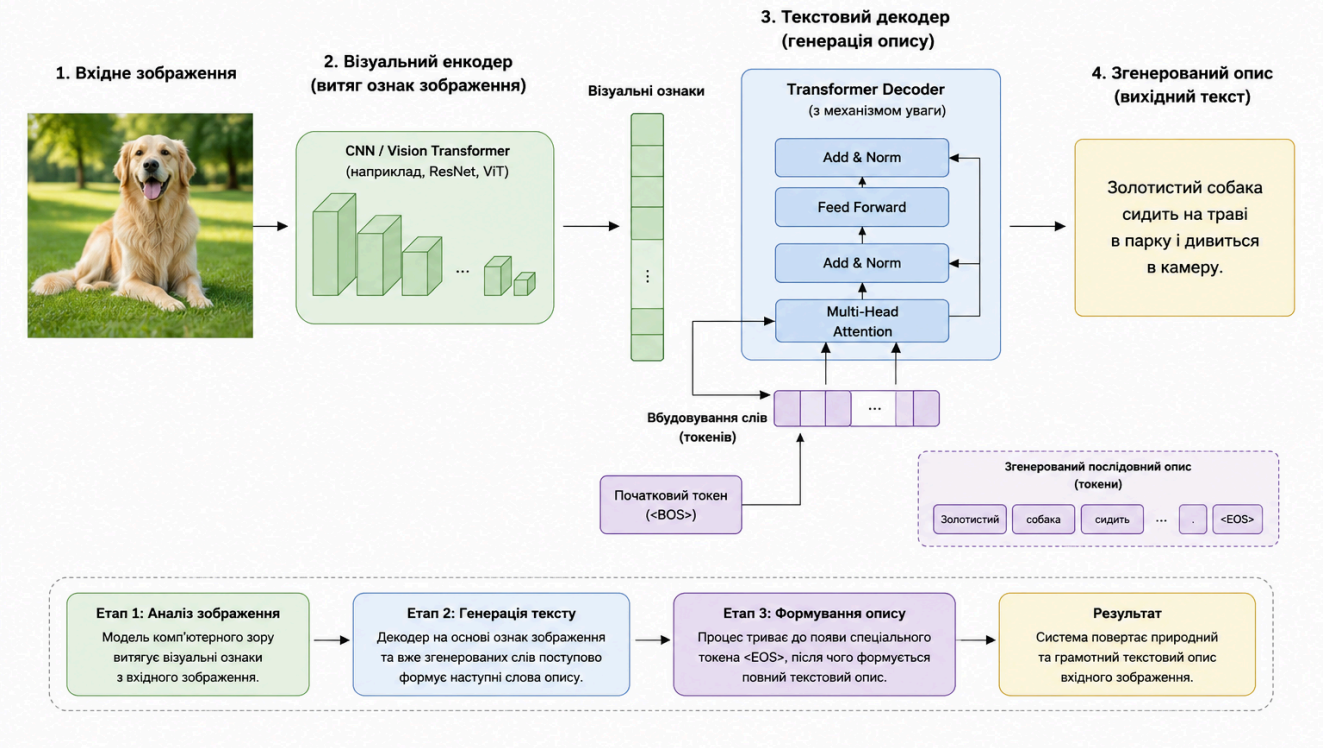

Рисунок 1.1 – Загальна схема роботи системи генерації текстових описів зображень

Сучасні Vision-Language моделі активно використовують Transformer-архітектури та механізми attention, які дозволяють ефективніше враховувати контекст сцени та взаємозв’язки між об’єктами зображення. Окремою особливістю задачі є те, що одне й те саме зображення може мати декілька коректних текстових описів, тому оцінювання якості моделі виконується за допомогою спеціалізованих метрик, таких як BLEU, ROUGE, METEOR та CIDEr.

Системи генерації текстових описів зображень використовуються у багатьох сферах інформаційних технологій. Одним із найбільш важливих напрямів є створення засобів доступності для людей із порушеннями зору, де автоматично сформований опис дозволяє отримати текстове або голосове представлення візуального контенту [3].

Технології image captioning також застосовуються у пошукових системах, цифрових бібліотеках та соціальних мережах для автоматичного тегування та структуризації мультимедійного контенту. Це дозволяє покращити пошук за зображеннями та спрощує обробку великих обсягів даних. Основні сфери застосування систем генерації описів зображень наведено в таблиці 1.1.

Таблиця 1.1 - Основні сфери застосування систем генерації описів зображень

Сфера застосування	**Приклади використання**
Засоби доступності	Формування текстових або голосових описів для людей із порушеннями зору
Пошукові системи	Автоматичне тегування зображень та покращення пошуку мультимедійного контенту
Соціальні мережі	Генерація підписів до зображень, модерація та аналіз контенту
Цифрові бібліотеки та архіви	Індексація та структуризація великих наборів зображень
Робототехніка	Аналіз навколишнього середовища автономними системами
Системи відеоспостереження	Автоматичний опис подій та об’єктів у кадрі
Мобільні застосунки	Генерація описів фотографій у режимі реального часу
Освітні платформи	Формування описів навчальних ілюстрацій та мультимедійних матеріалів
Електронна комерція	Автоматичне створення описів товарів на основі фотографій
AI-платформи та чат-боти	Використання мультимодальних моделей для взаємодії з користувачем

Крім цього, мультимодальні системи генерації описів використовуються у робототехніці, автономних системах, мобільних застосунках та AI-платформах. Такі моделі можуть бути інтегровані у вебсервіси або використовуватись як окремі модулі аналізу мультимедійного контенту.

Незважаючи на значний розвиток Vision-Language моделей, задача генерації текстових описів зображень все ще має низку проблем та обмежень. Однією з основних проблем є складність повного розуміння контексту сцени. Модель може правильно визначати окремі об’єкти, однак формувати неточні або надто загальні описи [4].

Сучасні моделі також потребують великих обсягів навчальних даних та значних обчислювальних ресурсів. Використання Transformer-архітектур супроводжується високими вимогами до GPU-пам’яті та часу навчання, що ускладнює використання таких моделей у системах із обмеженими ресурсами.

Додатковою проблемою є залежність якості моделі від структури та якості датасету. Наявність шумових анотацій або недостатня кількість прикладів для окремих категорій може негативно впливати на результати навчання. Основні проблеми та обмеження сучасних Vision-Language моделей наведено в таблиці 1.2.

Таблиця 1.2 - Основні проблеми та обмеження сучасних Vision-Language моделей

Проблема	**Опис**
Неповне розуміння контексту	Модель може некоректно визначати взаємозв’язки між об’єктами або неправильно інтерпретувати сцену
Генерація шаблонних описів	Система інколи формує загальні або повторювані текстові описи без деталізації
Високі вимоги до ресурсів	Навчання моделей потребує значних обчислювальних ресурсів, GPU-пам’яті та часу
Залежність від якості датасету	Помилки або шум у текстових анотаціях негативно впливають на результати навчання
Дисбаланс даних	Недостатня кількість прикладів окремих категорій знижує якість генерації описів
Обмежена багатомовність	Більшість моделей орієнтовані на англомовні набори даних
Складність опису рідкісних об’єктів	Модель може некоректно описувати незвичні сцени або малопоширені об’єкти
Недостатня узгодженість тексту	У деяких випадках опис може бути граматично правильним, але семантично неточним
Перенавчання моделі	При недостатньому обсязі даних модель може втрачати здатність до узагальнення
Складність інтеграції	Великі мультимодальні моделі можуть бути важкими для інтеграції у системи з обмеженими ресурсами

Більшість сучасних моделей також орієнтовані переважно на англомовні набори даних, що створює труднощі при побудові систем генерації описів українською мовою. Це робить актуальним подальший розвиток мультимодальних моделей та адаптацію Vision-Language підходів для багатомовних систем.

1.2 Аналіз сучасних Vision-Language моделей

Перші системи генерації текстових описів зображень базувалися на поєднанні згорткових нейронних мереж (CNN) та рекурентних нейронних мереж (RNN). У таких моделях CNN використовується для аналізу зображення та виділення візуальних ознак, а RNN або LSTM — для послідовної генерації тексту.

Найчастіше як візуальний енкодер використовувалися моделі ResNet або Inception, тоді як текстовий опис формувався за допомогою LSTM-декодера. Такий підхід дозволив автоматизувати генерацію описів та став основою для розвитку задачі image captioning [5].

Основною перевагою CNN-RNN моделей є відносно проста архітектура та невеликі вимоги до ресурсів. Проте рекурентні мережі мають обмеження при роботі з довгими залежностями та складними сценами, через що якість описів часто є недостатньо точною або шаблонною.

Подальший розвиток систем image captioning пов’язаний із використанням Transformer-архітектур та механізму attention. На відміну від RNN, Transformer дозволяє ефективніше враховувати взаємозв’язки між елементами сцени та формувати більш природні текстові описи.

У сучасних Vision-Language системах широко використовуються Vision Transformer (ViT) моделі, які обробляють зображення у вигляді набору патчів. Це дозволяє краще аналізувати глобальний контекст сцени та підвищує якість генерації тексту.

Transformer-підходи демонструють кращі результати у задачах image captioning порівняно з CNN-RNN моделями, однак потребують більших обчислювальних ресурсів та значних обсягів навчальних даних (рисунок 1.2).

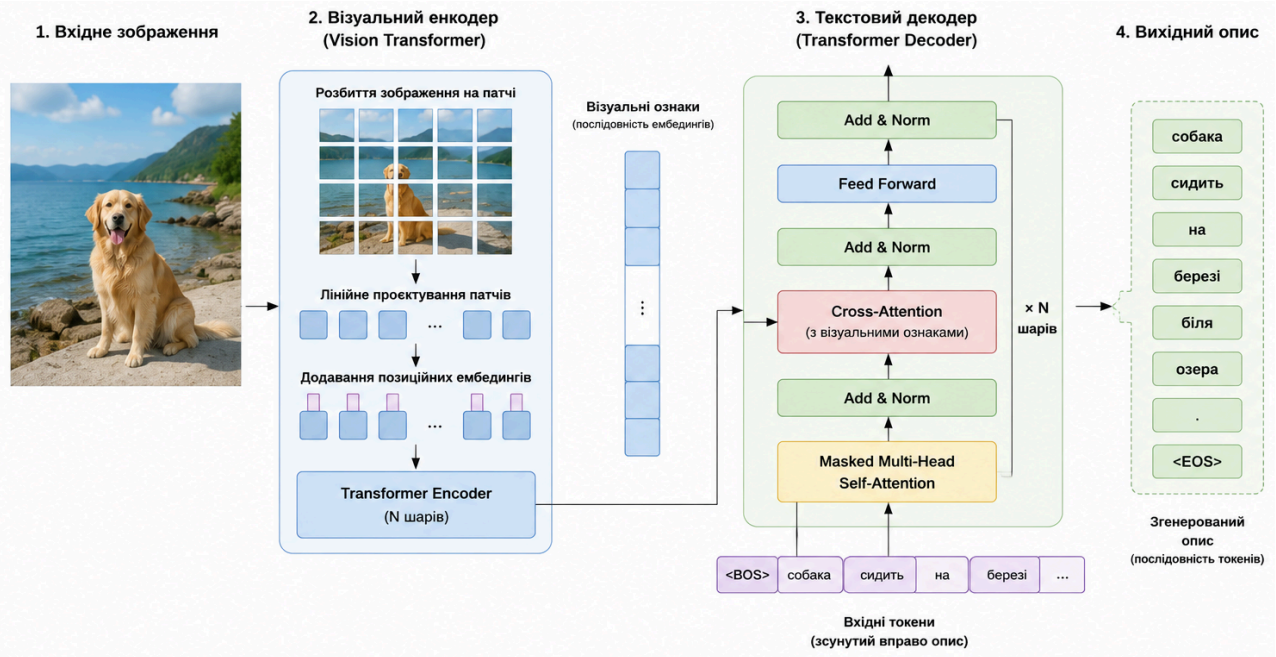

Рисунок 1.1 – Архітектура Transformer-моделі для задачі генерації описів зображень

Сучасні Vision-Language моделі поєднують аналіз зображень та генерацію тексту в межах єдиної мультимодальної архітектури. Одними з найбільш поширених моделей є BLIP, CLIP та Florence.

BLIP орієнтована на задачі генерації текстових описів та мультимодального навчання. Модель використовує Transformer-архітектуру та забезпечує високу якість генерації тексту завдяки попередньому навчанню на великих наборах даних.

CLIP використовується для узгодження текстових і візуальних представлень. Основною перевагою моделі є здатність працювати із zero-shot задачами та узагальнювати інформацію для нових категорій об’єктів.

Florence є мультимодальною моделлю компанії Microsoft, яка підтримує широкий спектр задач комп’ютерного зору та Vision-Language аналізу. Модель демонструє високу точність при роботі зі складними сценами та великими наборами даних [6].

Сучасні Vision-Language моделі відрізняються архітектурою, вимогами до ресурсів та якістю генерації тексту. CNN-RNN моделі мають простішу структуру,

однак поступаються Transformer-підходам у здатності враховувати контекст сцени. Transformer та мультимодальні моделі забезпечують більш природну генерацію тексту та кращу узгодженість між зображенням і текстом.

Таблиця 1.3 - Порівняння сучасних Vision-Language моделей

Модель	**Архітектура**	**Основне призначення**	**Переваги**	**Недоліки**
CNN-RNN	CNN + LSTM/GRU	Генерація описів зображень	Проста реалізація, невисокі вимоги до ресурсів	Слабке врахування глобального контексту, шаблонні описи
ViT + Transformer	Vision Transformer + Transformer Decoder	Image Captioning	Краще розуміння сцени та залежностей між об’єктами	Високі вимоги до GPU та даних
BLIP	Vision Transformer + Language Transformer	Генерація описів та мультимодальне навчання	Висока якість генерації тексту, попереднє навчання	Висока складність моделі
CLIP	Vision Encoder + Text Encoder	Узгодження тексту та зображення	Zero-shot можливості, хороше узагальнення	Не орієнтована безпосередньо на генерацію тексту
Florence	Multimodal Transformer	Комплексний Vision-Language аналіз	Висока точність та масштабованість	Великі обчислювальні витрати

Проведений аналіз показує, що найбільш ефективними для задачі генерації текстових описів зображень є Transformer та мультимодальні Vision-Language моделі. Вони забезпечують вищу точність генерації та краще працюють зі складними сценами, що робить їх найбільш доцільними для використання у сучасних системах image captioning.

1.3 Аналіз наборів даних для задачі генерації описів зображень

Для задачі генерації текстових описів зображень використовуються мультимодальні датасети, які містять зображення та відповідні текстові анотації. Найбільш поширеними наборами даних є MS COCO Captions, Flickr8k та

Flickr30k. Вони відрізняються кількістю зображень, складністю сцен та обсягом текстових описів [7].

У даній роботі основним набором даних обрано MS COCO Captions, оскільки він містить велику кількість різноманітних зображень, декілька описів для кожного прикладу та широко використовується у сучасних Vision-Language дослідженнях. Flickr8k та Flickr30k можуть використовуватись для додаткового тестування або адаптації моделі до інших наборів даних (рисунок 1.3).

Основні характеристики датасету MS COCO Captions:

- 118 000 зображень;
- 5 текстових описів для кожного зображення;
- формат анотацій JSON;
- 80 категорій об’єктів;
- реальні сцени з людьми, транспортом, тваринами та побутовими об’єктами;
- підтримка більшості сучасних ML-фреймворків.

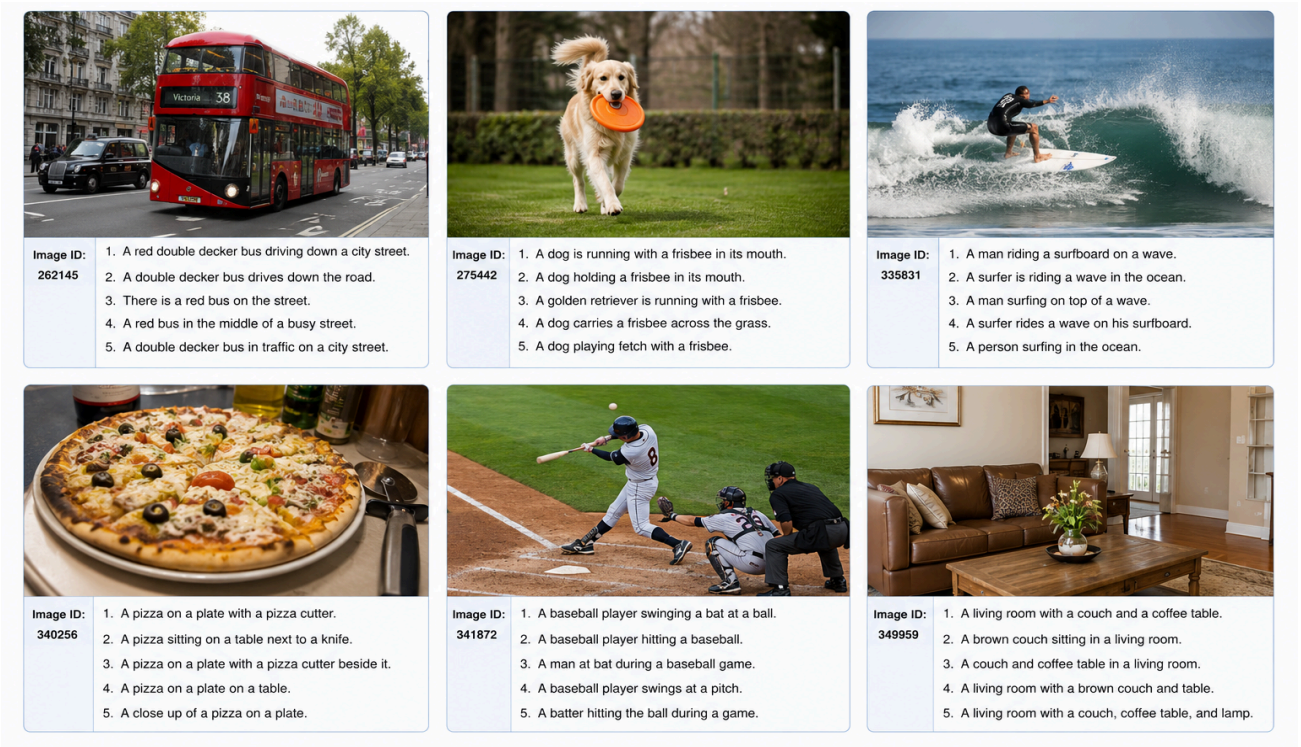

Рисунок 1.3 – Приклади зображень та текстових описів із датасету MS COCO Captions

Більшість сучасних датасетів для image captioning мають схожу структуру: зображення та набір текстових анотацій, які зберігаються у форматі JSON або CSV. Для одного зображення зазвичай доступно декілька описів, що дозволяє моделі враховувати різні варіанти формування тексту.

У MS COCO анотації містять ідентифікатор зображення, текстовий опис та додаткову інформацію про об’єкти сцени. Така структура є зручною для використання у PyTorch, TensorFlow та інших фреймворках машинного навчання (рисунок 1.4).

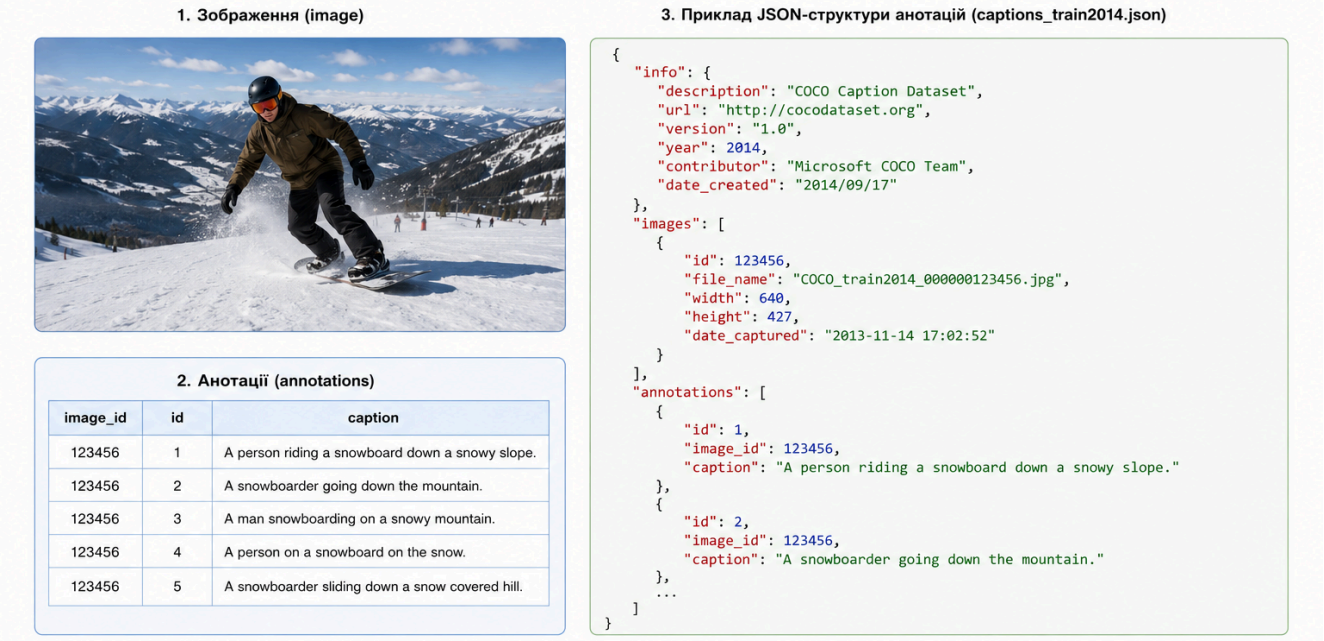

Рисунок 1.4 – Приклад структури анотацій датасету MS COCO

Основними проблемами підготовки датасетів є неоднорідність текстових описів, дисбаланс окремих категорій та наявність шумових анотацій. Частина об’єктів може зустрічатися значно частіше за інші, що впливає на якість навчання моделі. Перед навчанням також необхідно виконувати очищення тексту, токенізацію анотацій та попередню обробку зображень [8].

Стандартизована структура сучасних датасетів дозволяє легко адаптувати модель до інших наборів даних без суттєвих змін архітектури програмної системи.

1.4 Аналіз програмних засобів та бібліотек для реалізації системи

Для реалізації системи генерації текстових описів зображень використовуються сучасні фреймворки машинного навчання, бібліотеки обробки зображень та засоби роботи з текстом. Найбільш поширеними платформами для розробки Vision-Language моделей є PyTorch та TensorFlow. У даній роботі обрано PyTorch, оскільки він має гнучку архітектуру, зручний механізм роботи з нейронними мережами та широко використовується у сучасних дослідженнях мультимодальних моделей.

Для обробки зображень використовуються бібліотеки OpenCV та Pillow, які забезпечують зміну розмірів зображень, нормалізацію та підготовку даних перед навчанням моделі. Для роботи з текстовими анотаціями та токенізацією застосовуються бібліотеки NLTK, SpaCy та Hugging Face Transformers.

Навчання та тестування моделей виконується із використанням GPU-прискорення та сучасних інструментів глибинного навчання. Для оцінювання якості генерації описів використовуються метрики BLEU, ROUGE, METEOR та CIDEr. Використання сучасних бібліотек і стандартизованих інструментів дозволяє спростити процес розробки, навчання та подальшого масштабування Vision-Language системи.

1.5 Формування вимог до програмної системи

Програмна система генерації текстових описів зображень повинна забезпечувати автоматичний аналіз вхідного зображення та формування текстового опису природною мовою. Система має підтримувати роботу з Vision-Language моделями, виконувати попередню обробку даних та забезпечувати можливість подальшого масштабування або інтеграції з іншими сервісами [9].

1.5.1 Функціональні вимоги. Функціональні вимоги визначають основні можливості програмної системи та перелік операцій, які вона повинна виконувати. Для системи генерації текстових описів зображень ключовими є функції обробки вхідних даних, взаємодії з Vision-Language моделлю та формування текстового результату.

До основних функціональних вимог системи належать:

- завантаження зображення користувачем;
- попередня обробка зображення;
- генерація текстового опису;
- підтримка англомовних текстових описів;
- відображення результату генерації;
- можливість використання попередньо навченої моделі;
- підтримка тестування моделі на нових зображеннях.

Сформовані функціональні вимоги визначають базову логіку роботи програмної системи та забезпечують можливість автоматичного аналізу зображень і генерації текстових описів. Їх реалізація є основою для подальшого проєктування архітектури та реалізації програмного забезпечення.

1.5.2 Нефункціональні вимоги. Нефункціональні вимоги визначають характеристики якості програмної системи, продуктивність, стабільність та особливості її експлуатації. Для Vision-Language систем важливими є швидкість обробки даних, ефективність використання ресурсів та можливість масштабування моделі.

Система повинна забезпечувати:

- достатню швидкість генерації описів;
- стабільність роботи моделі;
- підтримку GPU-прискорення;
- сумісність із сучасними ML-бібліотеками;
- можливість масштабування системи;
- зручний програмний інтерфейс;

- коректну роботу з різними форматами зображень.

Дотримання нефункціональних вимог забезпечує стабільну та ефективну роботу системи генерації описів зображень. Це дозволяє використовувати програмне забезпечення як для локального тестування моделей, так і для подальшої інтеграції у вебсервіси або мультимодальні AI-платформи.

1.5.3 Сценарії використання системи. Сценарії використання системи визначають основні варіанти взаємодії користувача з програмним забезпеченням та описують ключові функції Vision-Language системи. На основі сформованих функціональних і нефункціональних вимог було визначено основні варіанти використання системи та побудовано UML-діаграму сценаріїв використання.

Основними акторами системи є:

- користувач;
- Vision-Language модель;
- система обробки зображень;
- модуль генерації тексту.

До основних сценаріїв використання системи належать:

- завантаження зображення;
- попередня обробка зображення;
- аналіз візуальних ознак;
- генерація текстового опису;
- відображення результату користувачу;
- тестування моделі на нових даних;
- використання попередньо навченої моделі.

UML-діаграму основних сценаріїв використання системи генерації текстових описів зображень наведено на рисунку 1.5.

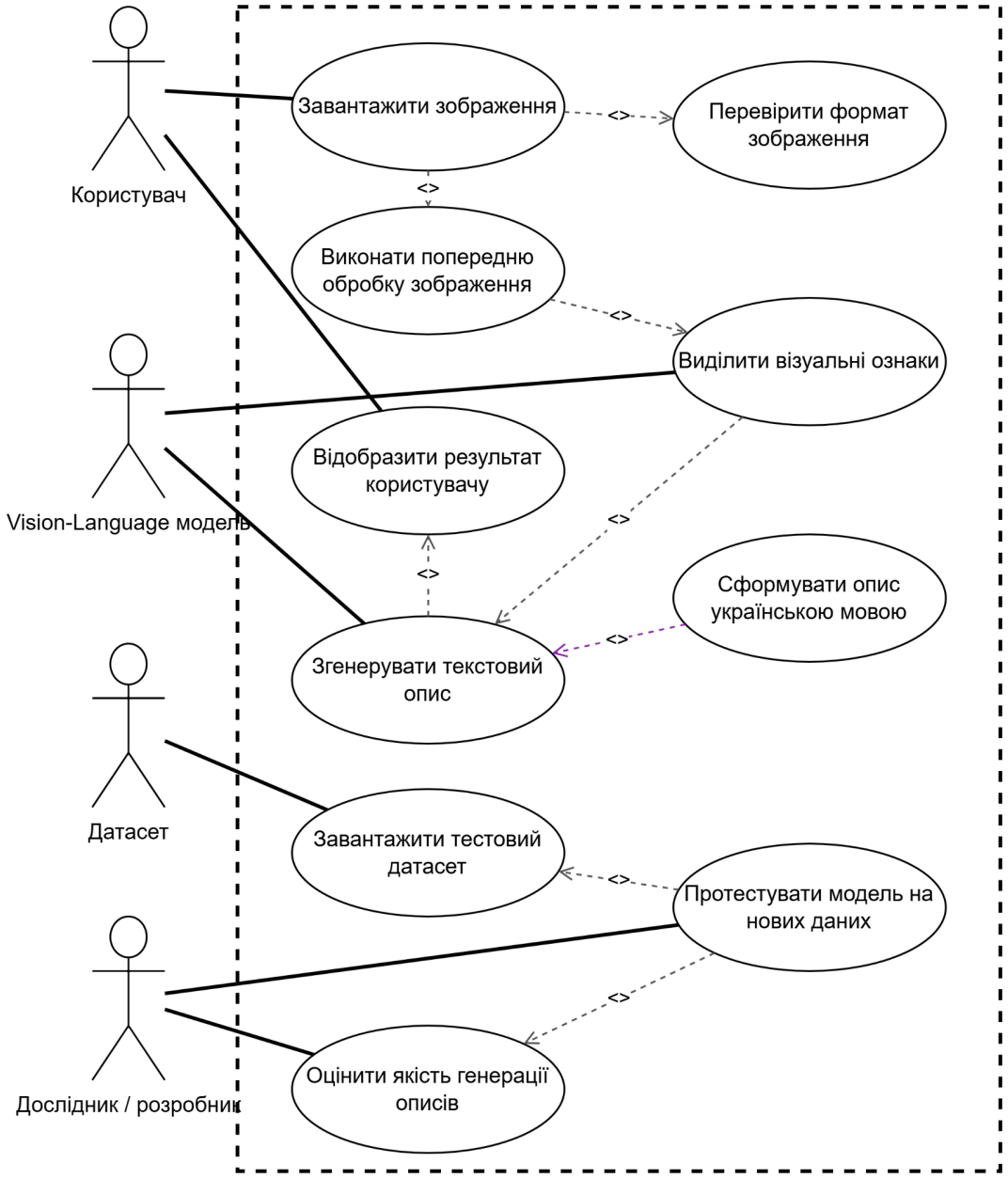

Рисунок 1.5 – UML-діаграма сценаріїв використання системи генерації текстових описів зображень

На основі функціональних і нефункціональних вимог було сформовано UML-діаграму варіантів використання системи генерації текстових описів зображень. Вона відображає основних акторів системи, їхню взаємодію з програмними модулями та послідовність виконання ключових функцій.

Основним актором є користувач, який завантажує зображення у систему та отримує сформований текстовий опис. Після завантаження зображення система виконує перевірку його формату, попередню обробку та передає дані до

Vision-Language моделі. Ці дії позначені зв’язками include, оскільки вони є обов’язковими етапами основного сценарію генерації опису.

Vision-Language модель взаємодіє з варіантами використання, пов’язаними з виділенням візуальних ознак і генерацією текстового опису. Спочатку модель аналізує вхідне зображення, визначає важливі візуальні ознаки, після чого текстовий декодер формує опис природною мовою. Результат передається користувачу через модуль відображення.

Окремим актором є дослідник або розробник, який виконує тестування моделі на нових даних та оцінює якість генерації описів. Для цього система може використовувати тестовий датасет, а також обчислювати метрики якості генерації. Такі сценарії потрібні для перевірки ефективності моделі та порівняння результатів її роботи.

Додатковим сценарієм є формування опису українською мовою. Він позначений зв’язком extend, оскільки не є обов’язковим для базового сценарію генерації, але може розширювати його функціональність. Основна модель може генерувати опис англійською мовою, а україномовний результат може формуватися як додатковий режим роботи системи.

Таким чином, UML-діаграма показує логіку роботи системи від завантаження зображення до отримання текстового опису, а також додаткові сценарії тестування, оцінювання якості та мовного розширення. Така структура дозволяє чітко розділити функції користувача, дослідника, датасету та Vision-Language моделі.

2. ПРОЄКТУВАННЯ ТА РОЗРОБКА ПРОГРАМНОЇ СИСТЕМИ ГЕНЕРАЦІЇ ОПИСІВ ЗОБРАЖЕНЬ

Після аналізу предметної області та сучасних Vision-Language підходів наступним етапом є проєктування архітектури програмної системи генерації текстових описів зображень. На цьому етапі визначаються основні компоненти системи, структура взаємодії між ними та архітектура моделі машинного навчання, яка використовується для задачі image captioning. Окрема увага приділяється побудові масштабованої структури системи, яка забезпечує можливість подальшого тестування, модифікації та інтеграції нових моделей або наборів даних [10].

2.1 Проєктування архітектури програмної системи

На основі сформованих функціональних вимог, нефункціональних вимог та UML-діаграми сценаріїв використання було сформовано загальну архітектуру програмної системи генерації текстових описів зображень. Архітектура системи включає модуль завантаження зображень, модуль попередньої обробки даних, Vision-Language модель, модуль генерації тексту та модуль відображення результатів користувачу [11].

Основною задачею архітектури є забезпечення послідовної обробки даних від моменту отримання зображення до формування текстового опису природною мовою. Загальна архітектура системи генерації описів зображень наведена на рисунку 2.1.

Архітектура програмної системи побудована за багаторівневим принципом, де кожен рівень відповідає за окрему частину функціональності системи:

- клієнтський рівень — забезпечує взаємодію користувача із системою, завантаження зображень та відображення результатів генерації;
- API-рівень — відповідає за обробку HTTP-запитів, перевірку даних та передачу інформації між компонентами системи;

- сервісний рівень — виконує попередню обробку зображень, керує процесом генерації описів та оцінюванням результатів;
- ML-рівень — містить Vision-Language модель, яка виконує аналіз зображення та генерацію текстового опису;
- рівень даних та журналювання — забезпечує зберігання ваг моделі, датасетів, логів роботи системи та результатів тестування.

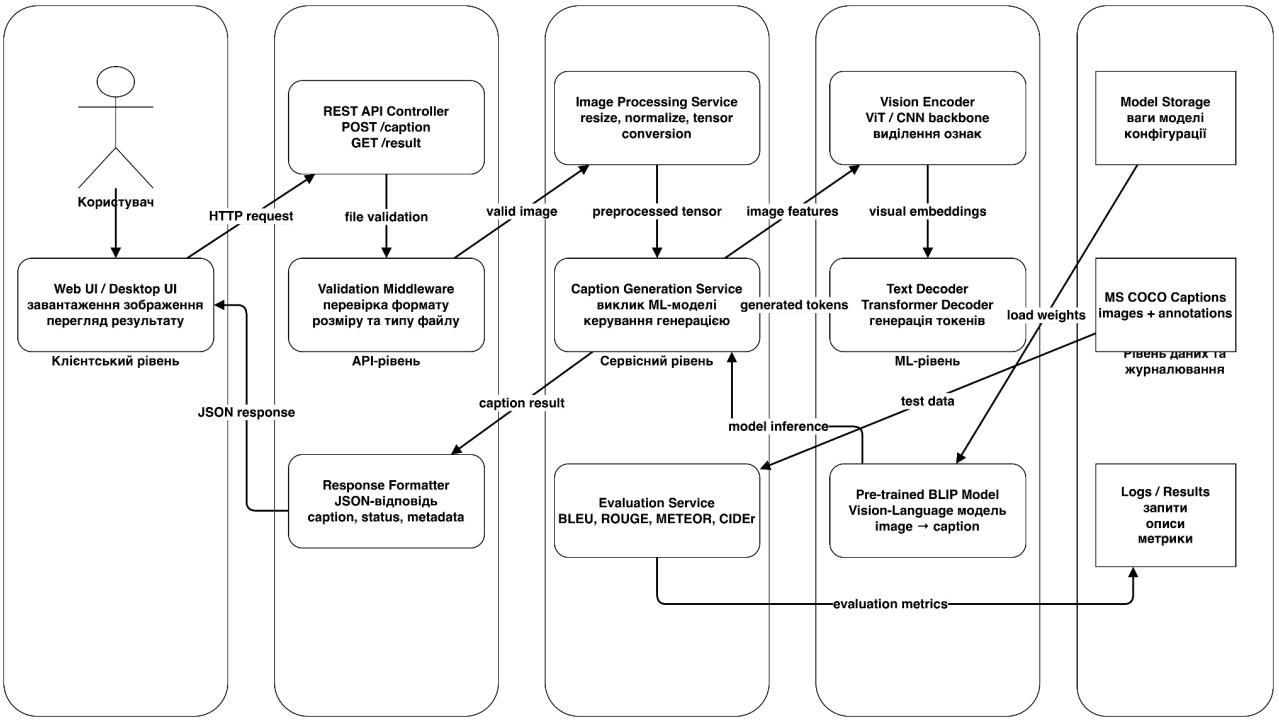

Рисунок 2.1 – Загальна архітектура системи генерації текстових описів зображень

Для реалізації системи генерації текстових описів зображень за основу було обрано Transformer-based Vision-Language модель BLIP. Дана модель поєднує Vision Transformer для аналізу зображення та Transformer-декодер для генерації тексту природною мовою [12].

Основною причиною вибору BLIP є висока якість генерації текстових описів, підтримка попереднього навчання на великих мультимодальних наборах даних та ефективна робота у задачах image captioning. Модель забезпечує краще

врахування контексту сцени та взаємозв’язків між об’єктами у порівнянні з класичними CNN-RNN підходами [13].

Архітектура моделі складається з візуального енкодера, який виконує аналіз вхідного зображення, та текстового декодера, який генерує послідовність текстових токенів. Для узгодження візуальних і текстових представлень використовуються механізми attention та Transformer-блоки. Архітектуру Vision-Language моделі BLIP наведено на рисунку 2.2.

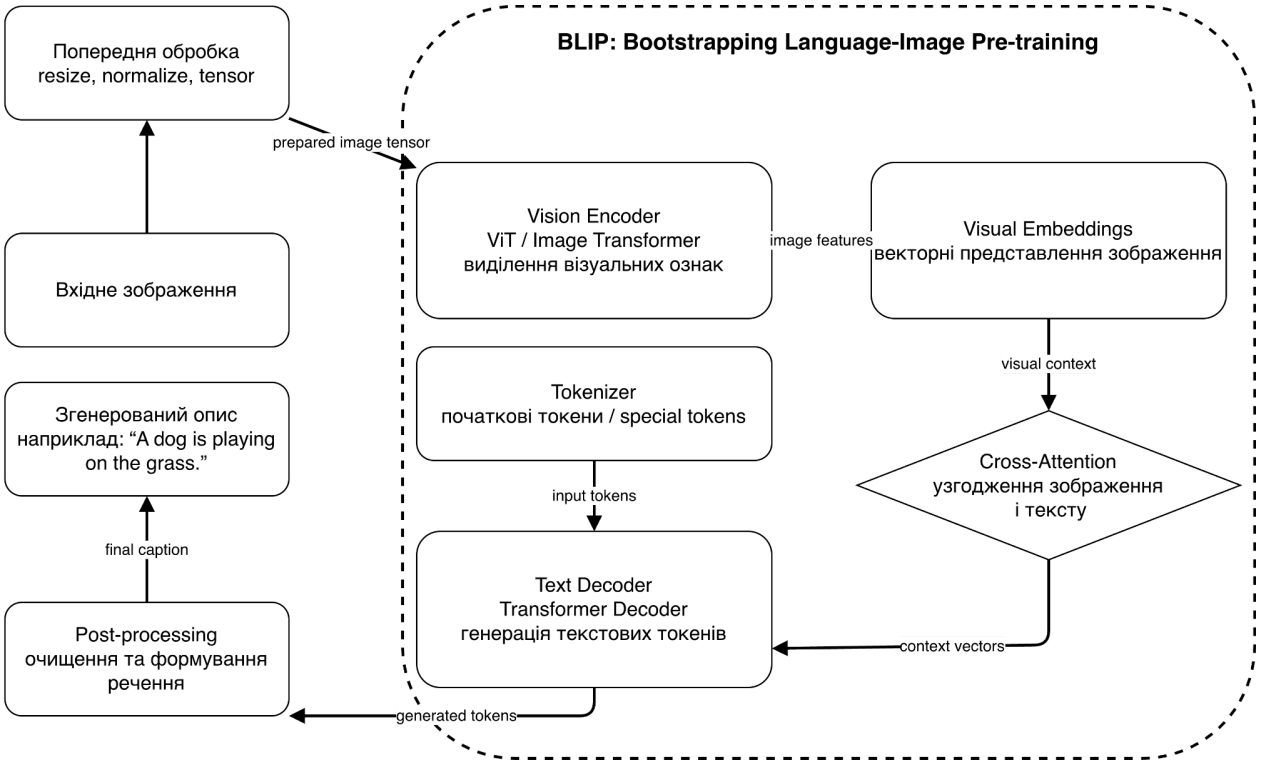

Рисунок 2.2 – Архітектура Vision-Language моделі BLIP

Основним компонентом програмної системи є Vision-Language модуль машинного навчання, який відповідає за аналіз зображення та генерацію текстового опису. Взаємодія компонентів системи побудована таким чином, щоб забезпечити послідовну передачу даних між клієнтським інтерфейсом, сервісними модулями та ML-моделлю. Після завантаження зображення користувачем дані передаються до модуля попередньої обробки, де виконується зміна розмірів

зображення, нормалізація та перетворення у формат тензорів для подальшої обробки моделлю [14].

Підготовлені дані надходять до Vision-Language моделі BLIP, яка складається з візуального енкодера та текстового декодера. Візуальний енкодер виділяє ознаки зображення та формує векторні представлення сцени, після чого текстовий декодер генерує послідовність текстових токенів. Для узгодження візуальних і текстових представлень використовуються механізми attention, що дозволяють моделі враховувати взаємозв'язки між об'єктами зображення та формувати більш точний опис [15].

Після завершення генерації тексту результат передається до сервісного модуля постобробки, де виконується формування кінцевого текстового опису та підготовка відповіді для користувача. Додатково система взаємодіє з модулем оцінювання якості, який використовує тестові дані та спеціалізовані метрики для аналізу ефективності моделі. Така організація взаємодії компонентів забезпечує модульність системи, спрощує масштабування та дозволяє інтегрувати нові Vision-Language моделі без суттєвих змін загальної архітектури програмного забезпечення.

2.2 Проєктування архітектури програмної системи

Якість роботи Vision-Language моделі значною мірою залежить від правильності підготовки даних та їх попередньої обробки. Перед навчанням моделі необхідно виконати завантаження датасету, обробку зображень, токенізацію текстових описів та формування навчальної, валідаційної і тестової вибірок. Дані етапи забезпечують коректну підготовку мультимодальних даних до навчання моделі та дозволяють зменшити вплив шуму, неоднорідності анотацій і різних форматів вхідних даних [16].

Для навчання Vision-Language моделі використовується датасет MS COCO Captions, який містить зображення та відповідні текстові описи у форматі JSON. Структура датасету включає набір зображень, анотації, ідентифікатори об'єктів та

текстові описи для кожного прикладу. Основні характеристики датасету наведено у таблиці 2.1.

Таблиця 2.1 - Основні характеристики датасету для навчання моделі

Характеристика	**Значення**
Датасет	MS COCO Captions
Тип даних	Зображення та текстові описи
Кількість зображень	118 000
Кількість описів	5 описів на зображення
Формат анотацій	JSON
Основна мова	Англійська
Кількість категорій	80
Тип сцен	Реальні повсякденні сцени
Призначення	Image Captioning

Перед передачею зображень до Vision-Language моделі виконується їх попередня обробка. Даний етап необхідний для приведення даних до єдиного формату та забезпечення коректної роботи моделі машинного навчання.

Основні етапи попередньої обробки зображень:

- завантаження зображення;
- зміна розміру зображення;
- нормалізація пікселів;
- перетворення у tensor-представлення;
- формування batch-наборів для навчання.

Загальну діаграма послідовності попередньої обробки та роботи з моделлю зображень наведено на рисунку 2.3.

Діаграма послідовності відображає узагальнений процес роботи системи генерації текстових описів зображень. Користувач завантажує зображення через інтерфейс системи, після чого запит передається до API-рівня. Backend виконує перевірку вхідних даних і передає їх до модуля обробки, де здійснюється підготовка зображення, нормалізація та формування службових токенів для моделі.

Після попередньої обробки дані передаються до Vision-Language моделі, яка виділяє візуальні ознаки зображення та генерує текстовий опис. Отриманий результат проходить через модуль формування відповіді та повертається користувачу у вигляді текстового опису. Повна деталізована діаграма послідовності наведена в додатку А.1.

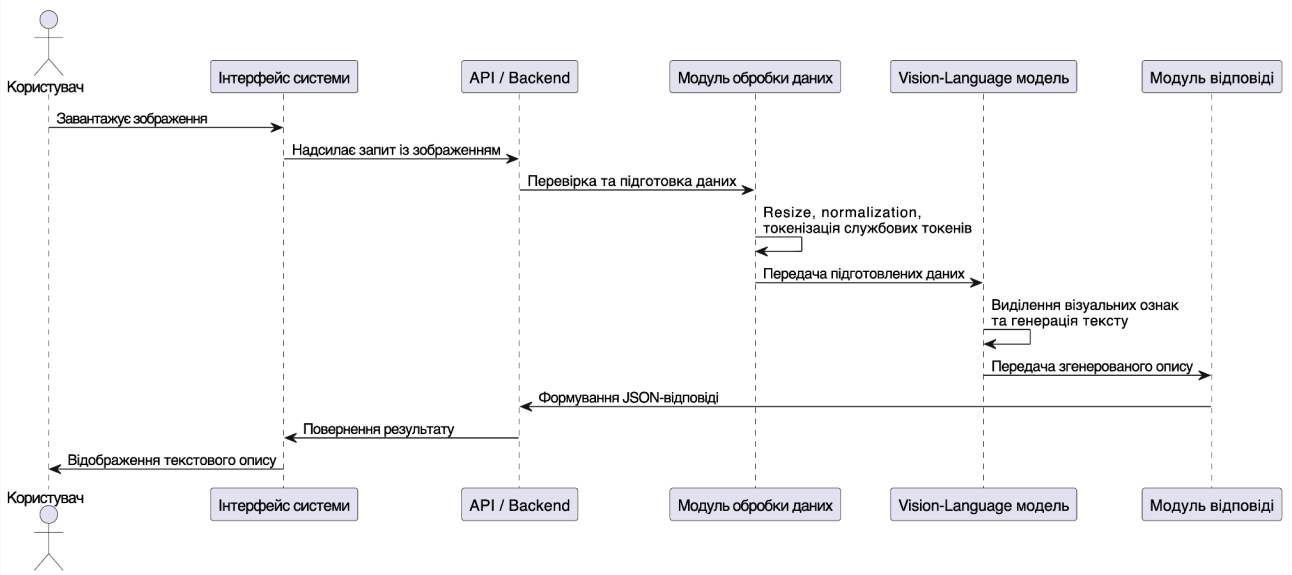

Рисунок 2.3 – Узагальнена діаграма послідовності роботи системи генерації описів зображень

Для обробки текстових описів використовується токенізація, яка дозволяє перетворити текст у послідовність токенів, придатних для подальшого використання Transformer-моделлю. Під час токенізації текст розбивається на окремі слова або підслова, після чого кожному токену присвоюється числовий ідентифікатор.

Додатково виконуються очищення тексту, приведення символів до єдиного формату та додавання спеціальних токенів початку й завершення речення. Це забезпечує коректне формування текстових послідовностей під час навчання та генерації описів [17].

Після завершення попередньої обробки дані поділяються на навчальну, валідаційну та тестову вибірки. Навчальна вибірка використовується для навчання

моделі, валідаційна — для контролю якості під час тренування, а тестова — для фінального оцінювання ефективності системи.

Таблиця 2.2 - Розподіл даних між навчальною, валідаційною та тестовою вибірками

Вибірка	**Кількість зображень**	**Частка**
Навчальна (Train)	82 000	70%
Валідаційна (Validation)	18 000	15%
Тестова (Test)	18 000	15%
Усього	118 000	100%

Для забезпечення коректного оцінювання моделі важливо уникати перетину даних між вибірками та зберігати різноманітність сцен і текстових описів у кожній частині датасету. Розподіл даних між вибірками наведено у таблиці 2.2.

2.3 Проєктування архітектури програмної системи

Реалізація Vision-Language моделі виконувалась із використанням Transformer-based архітектури BLIP, яка поєднує модулі аналізу зображень та генерації тексту в межах єдиної системи. Основними компонентами моделі є візуальний енкодер, текстовий декодер та механізм узгодження візуальних і текстових представлень. Загальну архітектуру моделі наведено на рисунку 2.2.

Для реалізації енкодера зображень використовувався Vision Transformer (ViT), який виконує поділ зображення на патчі та формує векторні представлення візуальних ознак. Такий підхід дозволяє ефективно враховувати глобальний контекст сцени та покращує якість генерації описів у порівнянні з класичними CNN-підходами. Перед передачею до енкодера зображення проходять етапи попередньої обробки, схема яких наведена на рисунку 2.3.

Текстовий декодер реалізований на основі Transformer Decoder та використовується для послідовної генерації текстових токенів. Для обробки текстових анотацій виконується токенізація, додавання спеціальних токенів

початку та завершення речення, а також формування словника моделі. Декодер генерує текстовий опис на основі візуальних ознак, отриманих від енкодера [18].

Інтеграція Vision та Language компонентів реалізована через механізм attention, який забезпечує узгодження текстових і візуальних представлень. Це дозволяє моделі враховувати взаємозв'язки між об'єктами сцени та формувати більш природні текстові описи. Загальний процес взаємодії компонентів системи наведено на рисунку 2.4.

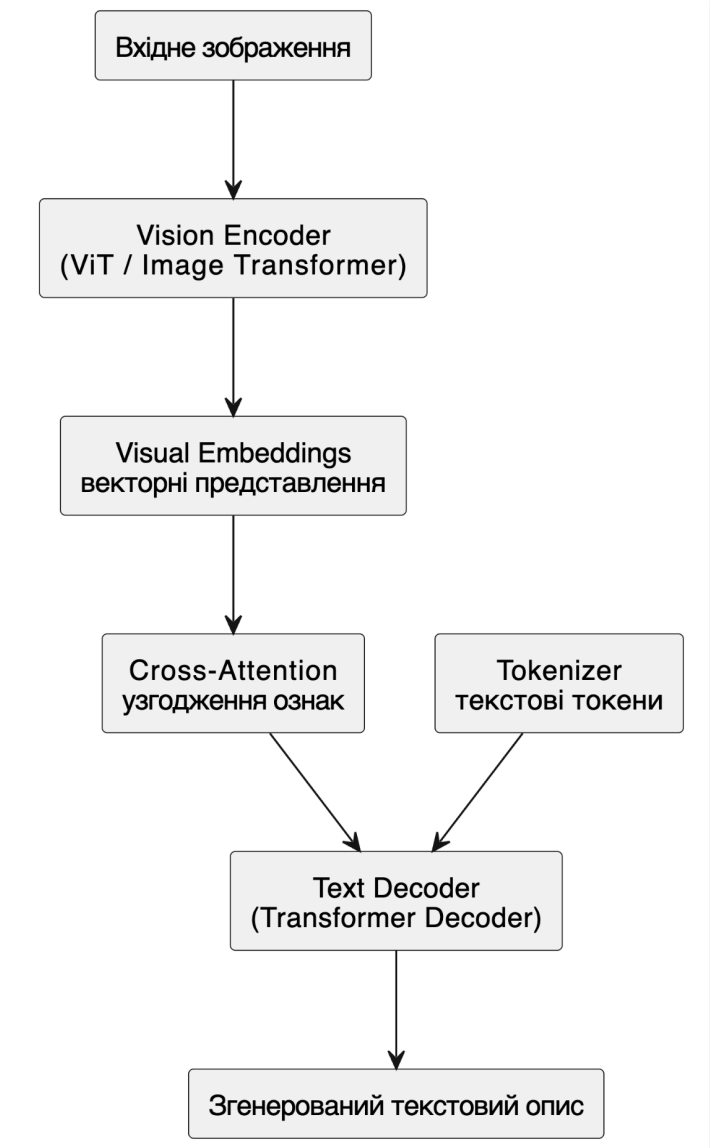

Рисунок 2.4 – Взаємодія Vision та Language компонентів моделі

Для навчання моделі використовувався датасет MS COCO Captions, характеристики якого наведено у таблиці 2.1, а розподіл даних між вибірками — у таблиці 2.2. Навчання виконувалось із використанням GPU-прискорення та оптимізатора AdamW. Основні параметри навчання моделі наведено у таблиці 2.3.

Таблиця 2.2 - Розподіл даних між навчальною, валідаційною та тестовою вибірками

Параметр	**Значення**
Модель	BLIP
Датасет	MS COCO Captions
Розмір зображення	224×224
Оптимізатор	AdamW
Функція втрат	Cross-Entropy Loss
Batch size	32
Кількість епох	10
Learning rate	1e-4
Пристрій навчання	GPU
Метрики оцінювання	BLEU, ROUGE, METEOR, CIDEr
Фреймворк	PyTorch

Під час навчання використовувались механізми batch-обробки, нормалізації даних та регуляризації для зменшення перенавчання моделі. Оцінювання якості генерації текстових описів виконувалось із використанням метрик BLEU, ROUGE, METEOR та CIDEr. Це дозволило забезпечити стабільну роботу моделі та отримати узгоджені результати генерації описів зображень.

2.4 Класове представлення для реалізація програмної системи

Реалізація програмної системи передбачає створення трьох основних частин: серверної логіки, користувацького інтерфейсу та API для взаємодії з моделлю машинного навчання. Для реалізації доцільно використати Python, PyTorch, FastAPI, React та бібліотеки обробки зображень.

Серверна частина відповідає за прийом зображення, перевірку вхідних даних, попередню обробку, виклик Vision-Language моделі та повернення

результату. Для реалізації серверної логіки доцільно використати Python, FastAPI, PyTorch, Pillow/OpenCV та Hugging Face Transformers (рисунок 2.5).

Основні технології:

- Python — основна мова реалізації;
- FastAPI — створення серверної частини та REST API;
- PyTorch — робота з моделлю машинного навчання;
- Pillow / OpenCV — обробка зображень;
- Transformers — завантаження та використання BLIP-моделі.

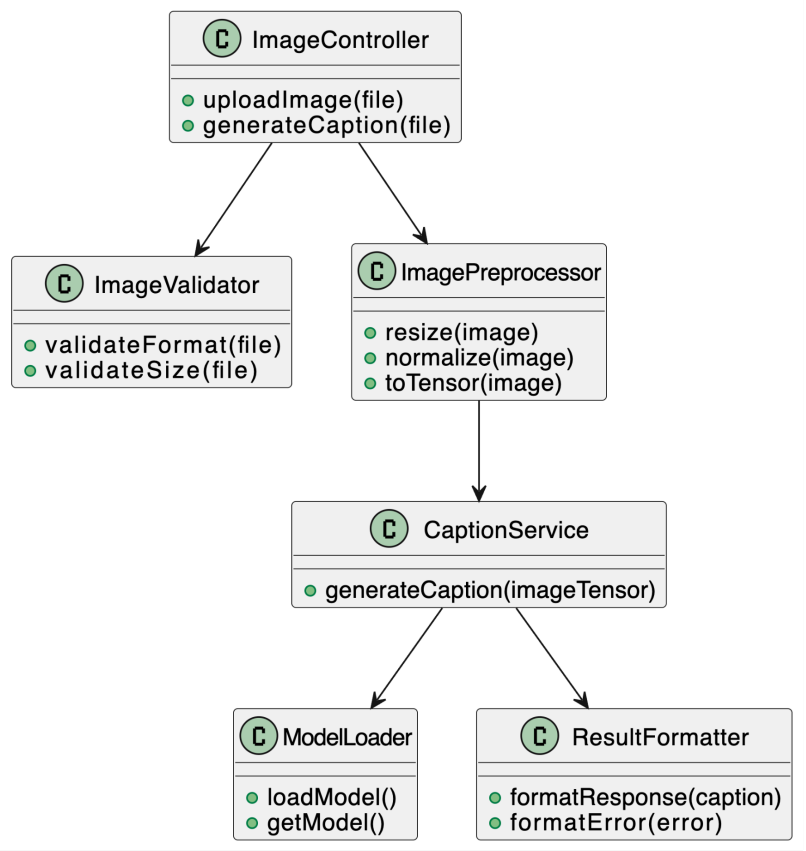

Рисунок 2.5 – Діаграма класів серверної частини системи

Інтерфейс користувача забезпечує завантаження зображення, надсилання його на сервер та відображення згенерованого опису. Для реалізації клієнтської частини доцільно використати React, TypeScript, HTML, CSS та Axios для взаємодії з API (рисунок 2.6).

Основні технології:

- React — побудова інтерфейсу користувача;
- TypeScript — типізація логіки клієнтської частини;
- Axios — надсилання HTTP-запитів;
- HTML/CSS — структура та стилізація інтерфейсу.

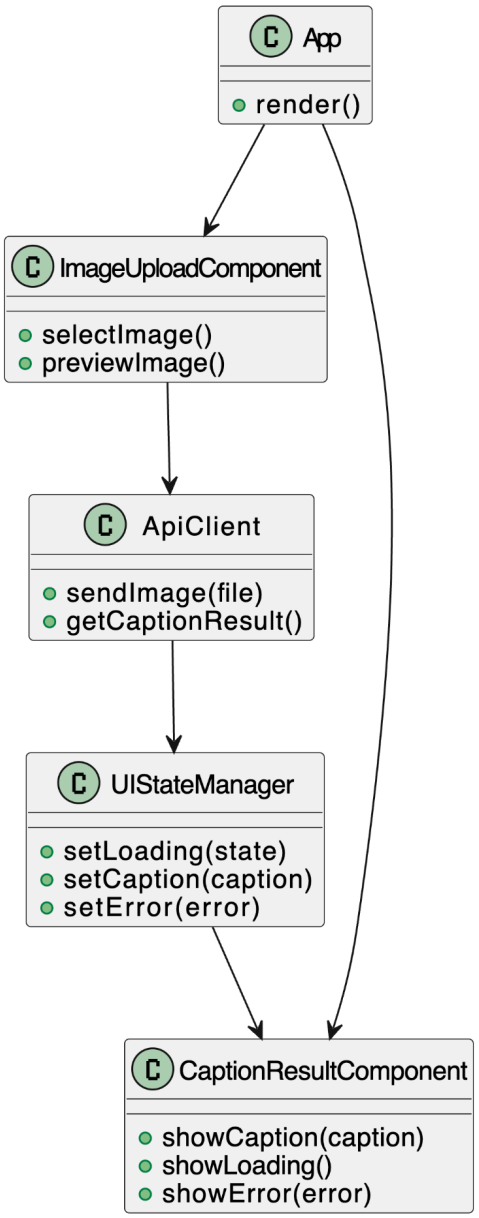

Рисунок 2.6 – Діаграма класів інтерфейсу користувача

API забезпечує зв’язок між інтерфейсом користувача та модулем машинного навчання. Основним endpoint є запит для передавання зображення та отримання

текстового опису. API також має виконувати перевірку файлу, обробку помилок і формування відповіді у форматі JSON (рисунок 2.7).

Основні технології:

- FastAPI — створення REST API;
- Pydantic — валідація даних;
- Uvicorn — запуск серверного застосунку;
- JSON — формат відповіді API.

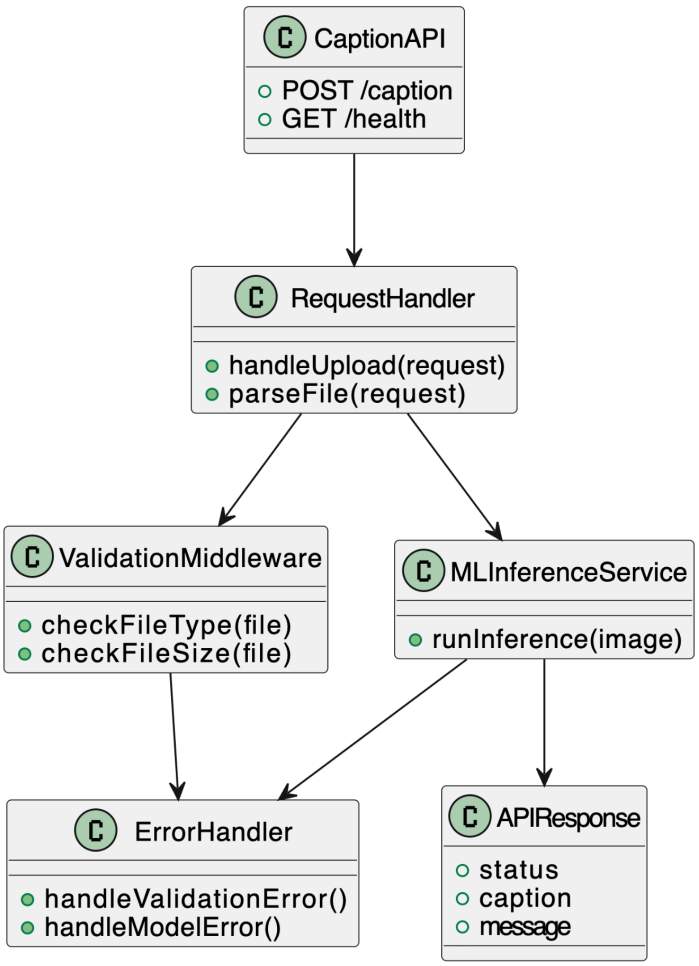

Рисунок 2.7 – Діаграма класів API взаємодії з Vision-Language моделлю

Реалізація програмної системи дозволила сформувати повноцінну архітектуру для генерації текстових описів зображень на основі Vision-Language підходів. Використання сучасних технологій, зокрема FastAPI, PyTorch, React та Transformer-моделей, забезпечило можливість ефективної взаємодії між

клієнтською частиною, серверною логікою та модулем машинного навчання. Запропонована структура системи є модульною та масштабованою, що дозволяє у подальшому інтегрувати нові моделі, розширювати функціональність програмного забезпечення та адаптувати систему до інших мультимодальних задач.

2.5 Реалізація процесу навчання моделі

Процес навчання Vision-Language моделі реалізовано із використанням фреймворку PyTorch та попередньо навченої BLIP-моделі. Навчання виконується на підготовленому датасеті MS COCO Captions із використанням розподілу даних на навчальну, валідаційну та тестову вибірки, наведеного у таблиці 2.2. Основні параметри навчання моделі наведено у таблиці 2.3.

Для конфігурації процесу навчання використовуються параметри batch size, learning rate, кількість епох та функція втрат Cross-Entropy Loss. Оптимізація ваг моделі виконується за допомогою оптимізатора AdamW, який забезпечує стабільніше навчання Transformer-архітектур та зменшує ризик перенавчання. Під час тренування використовується batch-обробка даних та автоматичне перемішування вибірки для покращення узагальнювальної здатності моделі [19].

Навчання моделі виконується із використанням GPU-прискорення, що дозволяє суттєво зменшити час обробки великих мультимодальних даних. Для оптимізації обчислень використовуються tensor-операції PyTorch, механізми автоматичного обчислення градієнтів та підтримка паралельної обробки batch-наборів. Додатково застосовується нормалізація даних та кешування частини оброблених зображень для зменшення навантаження на систему під час тренування.

Після завершення навчання виконується збереження ваг моделі, параметрів конфігурації та результатів оцінювання. Для цього використовуються файли checkpoint, які дозволяють повторно завантажувати модель без повторного навчання. Збереження проміжних станів моделі також дає можливість відновлення процесу навчання у випадку помилки або переривання роботи системи.

Схему процесу навчання Vision-Language моделі наведено на рисунку 2.8. На початковому етапі система завантажує дані з датасету MS COCO Captions, після чого виконується попередня обробка зображень та текстових анотацій. Далі формується розподіл даних на навчальну, валідаційну та тестову вибірки, які використовуються для тренування та оцінювання моделі. Після підготовки дані передаються до Vision-Language моделі BLIP, де виконується навчання із використанням GPU-прискорення та механізмів backpropagation. Після завершення тренування система обчислює метрики якості генерації текстових описів, зокрема BLEU, ROUGE, METEOR та CIDEr, а також виконує збереження checkpoint-моделі для подальшого використання або повторного завантаження.

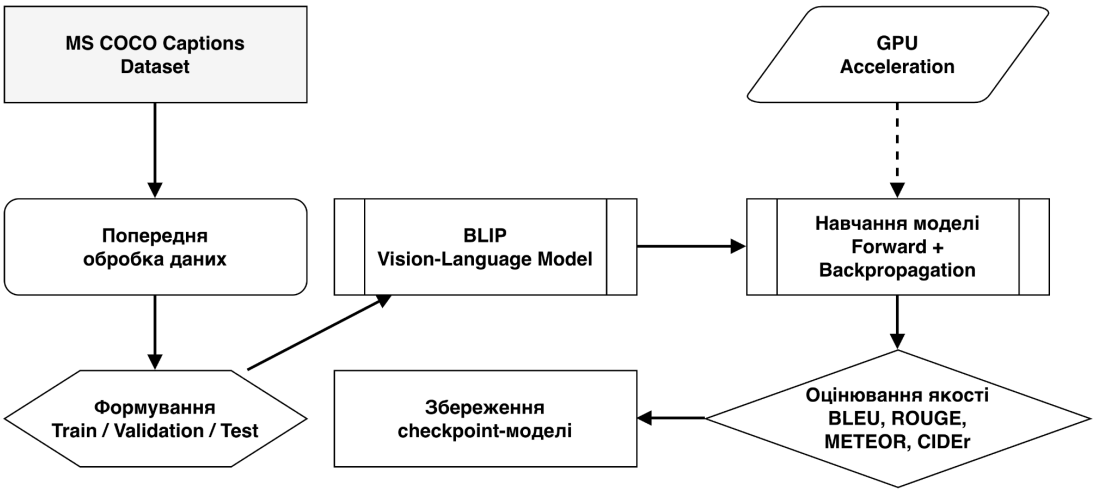

Рисунок 2.8 – Схема процесу навчання Vision-Language моделі

Для оцінювання якості моделі використовуються метрики BLEU, ROUGE, METEOR та CIDEr, які дозволяють аналізувати точність та узгодженість згенерованих текстових описів. Такий підхід забезпечує контроль ефективності моделі на різних етапах навчання та дозволяє виконувати подальше покращення параметрів системи.

2.6 Демонстрація роботи програмної системи

Для перевірки працездатності розробленої системи було виконано тестування основних функціональних можливостей програмного забезпечення, зокрема процесу навчання моделі, генерації текстових описів та взаємодії користувача із системою. Демонстрація роботи програмної системи дозволяє оцінити коректність реалізації Vision-Language моделі та перевірити взаємодію між основними компонентами архітектури [20].

На рисунку 2.9 наведено приклад процесу навчання Vision-Language моделі. У процесі навчання система виконує обробку batch-наборів даних, обчислення функції втрат та оновлення ваг моделі. Також відображаються основні параметри тренування та проміжні результати оцінювання моделі.

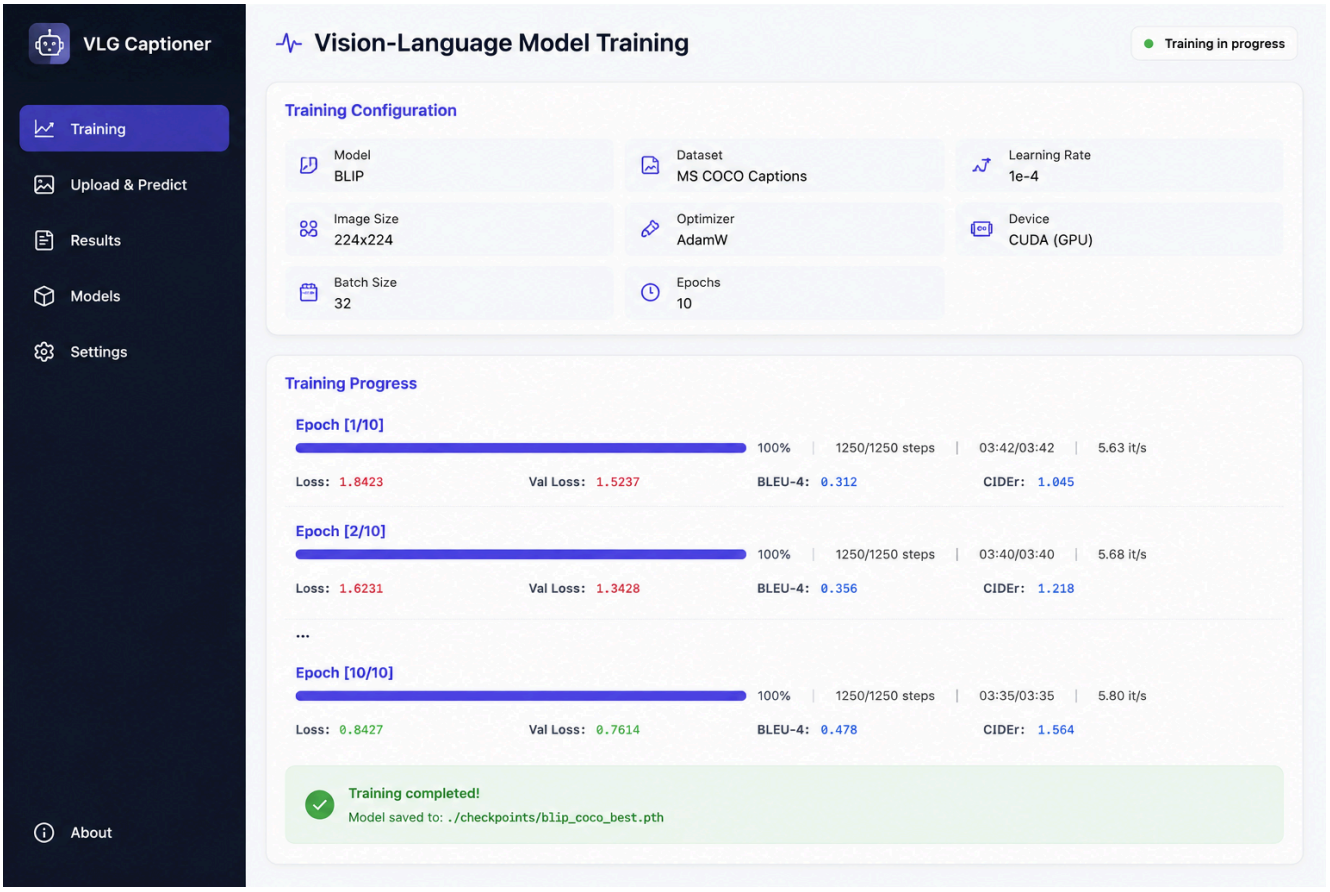

Рисунок 2.9 – Процес навчання Vision-Language моделі

На рисунку 2.10 наведено приклад роботи користувацького інтерфейсу системи під час завантаження зображення для генерації текстового опису. Користувач завантажує вхідне зображення через інтерфейс системи, після чого дані передаються до Vision-Language моделі для подальшого аналізу.

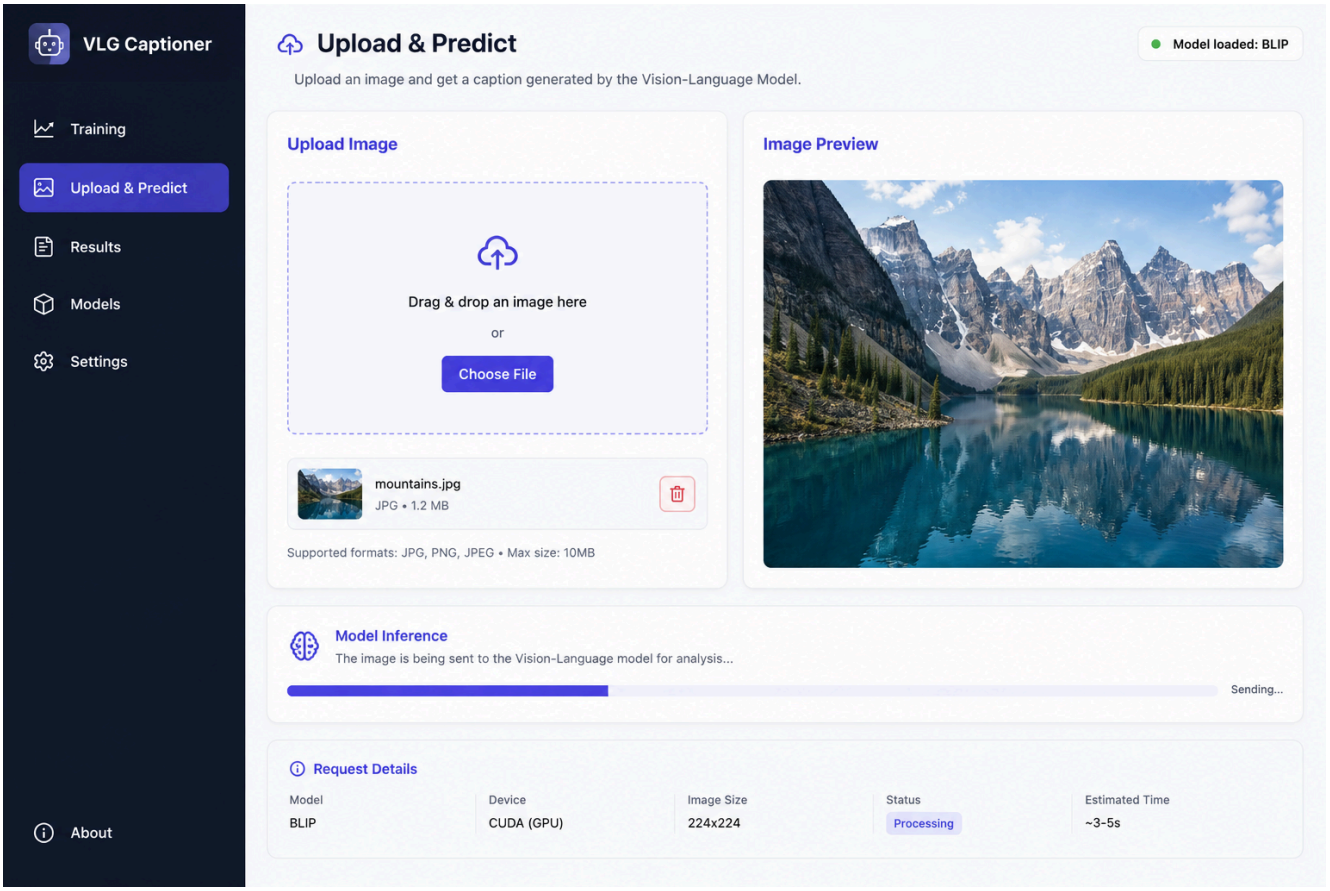

Рисунок 2.10 – Завантаження зображення у програмній системі

Результат роботи системи наведено на рисунку 2.11. Після аналізу зображення модель генерує текстовий опис природною мовою та повертає його користувачу через інтерфейс системи. Отриманий результат демонструє здатність Vision-Language моделі формувати змістовні текстові описи на основі вхідних візуальних даних.

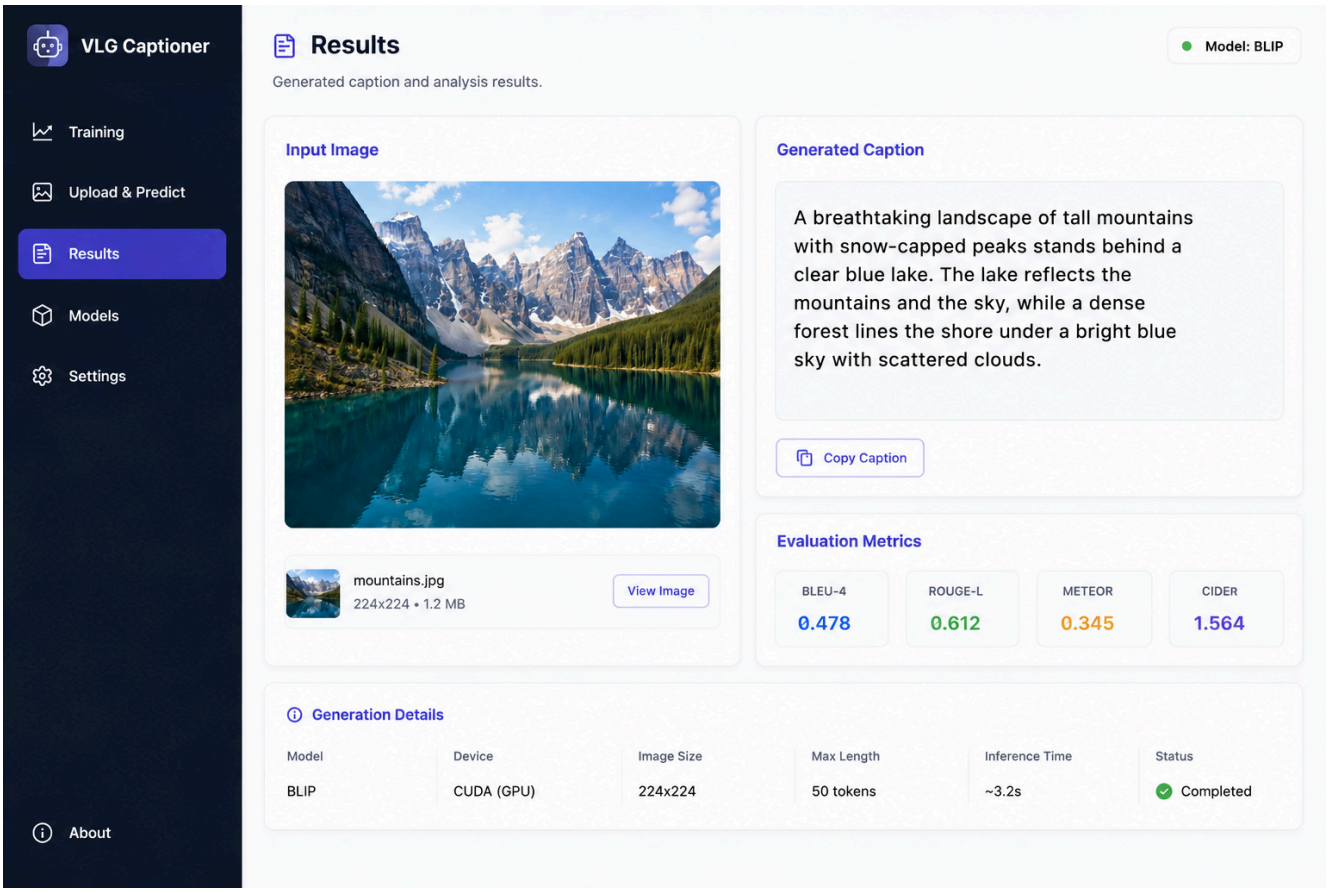

Рисунок 2.11 – Результат генерації текстового опису зображення

На рисунку 2.12 наведено приклад сторінки керування моделями у програмній системі генерації текстових описів зображень. Даний модуль дозволяє переглядати список доступних Vision-Language моделей, інформацію про датасети, на яких вони були навчені, значення основних метрик оцінювання та статус активності моделі. Для кожної моделі також відображаються дата створення, тип архітектури та параметри конфігурації. Користувач може обрати необхідну модель для подальшого використання у задачах генерації описів зображень. Реалізація такого модуля забезпечує зручне керування різними версіями моделей та спрощує процес тестування й порівняння результатів роботи системи.

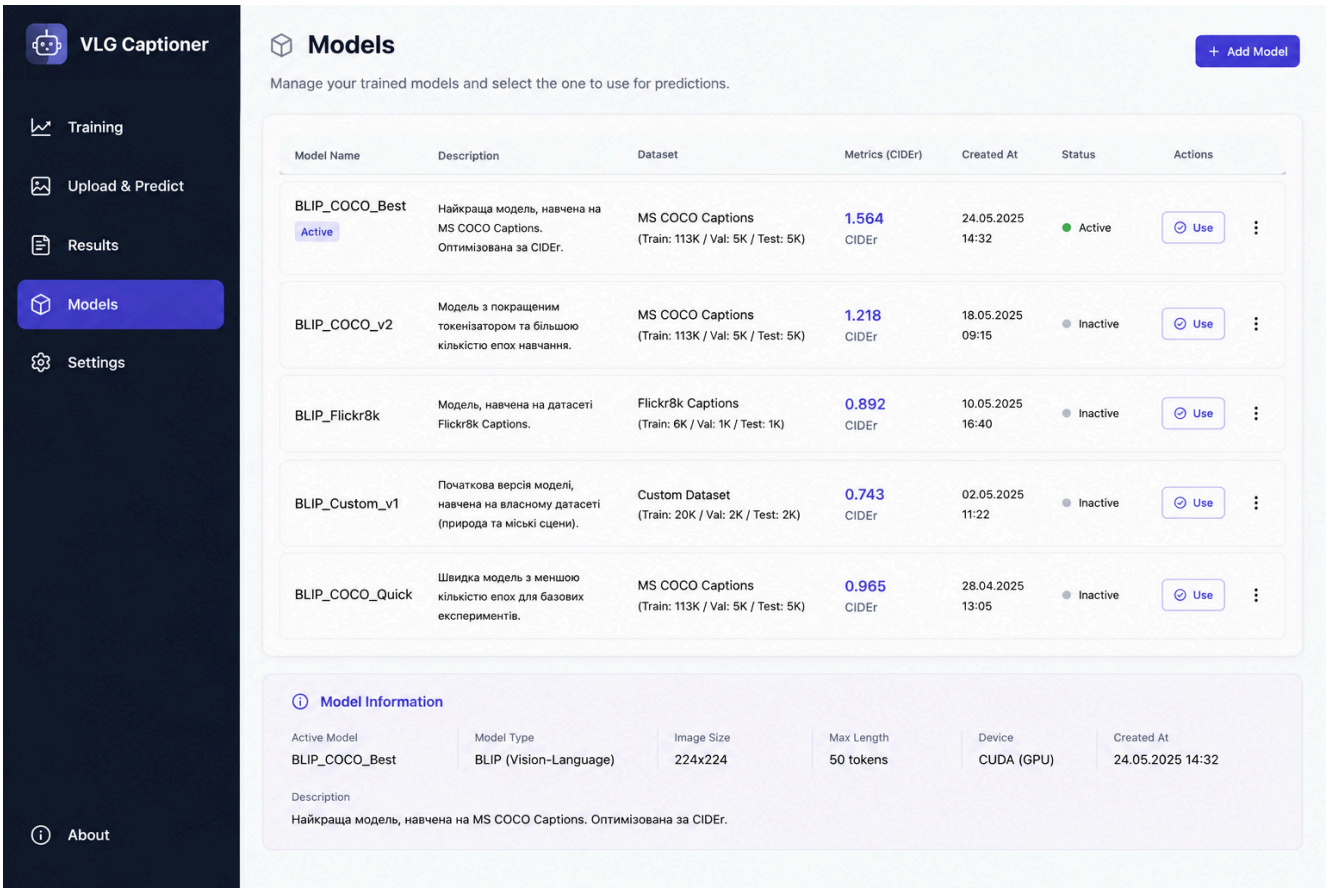

Рисунок 2.12 – Інтерфейс керування Vision-Language моделями у програмній системі

2.7 Висновки до 2 розділу

У межах другого розділу було виконано проєктування та реалізацію програмної системи генерації текстових описів зображень на основі Vision-Language підходів. Розроблено багаторівневу архітектуру системи, яка включає користувацький інтерфейс, серверну частину, API взаємодії та модуль машинного навчання. Це забезпечило модульність системи та можливість її подальшого розширення. Було реалізовано процес підготовки та попередньої обробки даних, механізми токенізації текстових описів, навчання Vision-Language моделі та збереження checkpoint-моделей. Для навчання використано датасет MS COCO Captions і GPU-прискорення, що дозволило оптимізувати обчислення та підвищити ефективність тренування моделі.

3. ТЕСТУВАННЯ, ОЦІНКА ЕФЕКТИВНОСТІ ТА ВПРОВАДЖЕННЯ СИСТЕМИ

Розроблення моделі машинного навчання для генерації текстових описів зображень потребує не лише створення та навчання архітектури, а й комплексної перевірки її працездатності, якості генерації та можливостей практичного використання. Важливим етапом дослідження є тестування програмної системи, оцінювання ефективності роботи моделі за відповідними метриками та аналіз отриманих результатів. Це дозволяє визначити рівень відповідності системи поставленим вимогам, виявити її сильні та слабкі сторони, а також оцінити перспективи використання у реальних умовах. У даному розділі наведено результати тестування розробленої системи, проведено оцінювання якості генерації текстових описів зображень, проаналізовано продуктивність моделі та розглянуто можливості її подальшого впровадження.

3.1 Організація експериментального дослідження

Після реалізації програмної системи було проведено тестування Vision-Language моделі, оцінювання якості генерації текстових описів та перевірку працездатності основних компонентів системи. Основною метою даного етапу є визначення ефективності моделі, аналіз якості згенерованих описів та оцінювання стабільності роботи програмного забезпечення в умовах обробки реальних зображень.

Для проведення експериментального дослідження використовувався датасет MS COCO Captions, який містить зображення та відповідні текстові анотації. Під час експериментів виконувалось навчання Vision-Language моделі, тестування генерації описів та оцінювання результатів за допомогою спеціалізованих метрик якості. Окрема увага приділялась перевірці коректності генерації тексту, стабільності роботи моделі та швидкості обробки вхідних даних.

Методика експериментального дослідження передбачала послідовне виконання етапів підготовки даних, навчання моделі, генерації текстових описів та оцінювання отриманих результатів. Для тестування використовувались зображення із тестової вибірки, які не брали участі у процесі навчання моделі.

Оцінювання ефективності системи виконувалось із використанням таких метрик:

- BLEU — оцінювання схожості згенерованого тексту з еталонними описами;
- ROUGE — оцінювання повноти та збігу текстових фрагментів;
- METEOR — аналіз семантичної та лексичної подібності тексту;
- CIDEr — оцінювання узгодженості опису відносно набору еталонних анотацій.

Додатково аналізувались швидкість генерації опису та стабільність роботи системи під час обробки різних типів зображень.

Навчання та тестування Vision-Language моделі виконувалось у середовищі Python із використанням бібліотек PyTorch, Transformers та OpenCV. Для реалізації серверної частини використовувався FastAPI, а для клієнтського інтерфейсу — React.

Обчислення виконувались із використанням GPU-прискорення, що дозволило зменшити час навчання та оптимізувати процес генерації текстових описів. Основні параметри апаратного та програмного середовища наведено у таблиці 3.1.

Таблиця 3.1 - Конфігурація апаратного та програмного середовища

Компонент	**Характеристика**
1	**2**
Операційна система	Windows 11
Мова програмування	Python 3.11
ML-фреймворк	PyTorch
Vision-Language бібліотека	Hugging Face Transformers
Серверна частина	FastAPI
Клієнтська частина	React

Продовження таблиці 3.1

Бібліотеки обробки зображень	OpenCV, Pillow
GPU	NVIDIA GeForce RTX 4060
CPU	Intel Core i7
Оперативна пам’ять	16 GB RAM
Середовище розробки	Visual Studio Code
Формат датасету	JSON та JPEG

3.2 Оцінювання якості генерації текстових описів

Оцінювання якості генерації текстових описів виконувалось на тестовій вибірці датасету MS COCO Captions із використанням спеціалізованих метрик для Vision-Language моделей. Основною метою оцінювання було визначення точності генерації описів, узгодженості тексту з вмістом зображення та порівняння ефективності різних архітектур моделей.

Для аналізу якості генерації використовувались метрики BLEU, ROUGE, METEOR та CIDEr. BLEU дозволяє оцінити схожість згенерованого тексту з еталонними описами на рівні n-грам. ROUGE використовується для аналізу повноти текстового опису, METEOR враховує семантичну та лексичну подібність, а CIDEr орієнтований саме на задачі image captioning та оцінює узгодженість опису із набором еталонних анотацій.

Для оцінювання ефективності системи було проведено порівняння декількох підходів генерації текстових описів, зокрема CNN-RNN архітектури, Transformer-based моделі та Vision-Language моделі BLIP. Отримані результати показали, що Transformer-підходи забезпечують вищу якість генерації тексту та краще враховують контекст сцени у порівнянні з класичними CNN-RNN моделями.

Результати порівняння моделей за основними метриками наведено у таблиці 3.2, а графічне порівняння ефективності моделей — на рисунку 3.1.

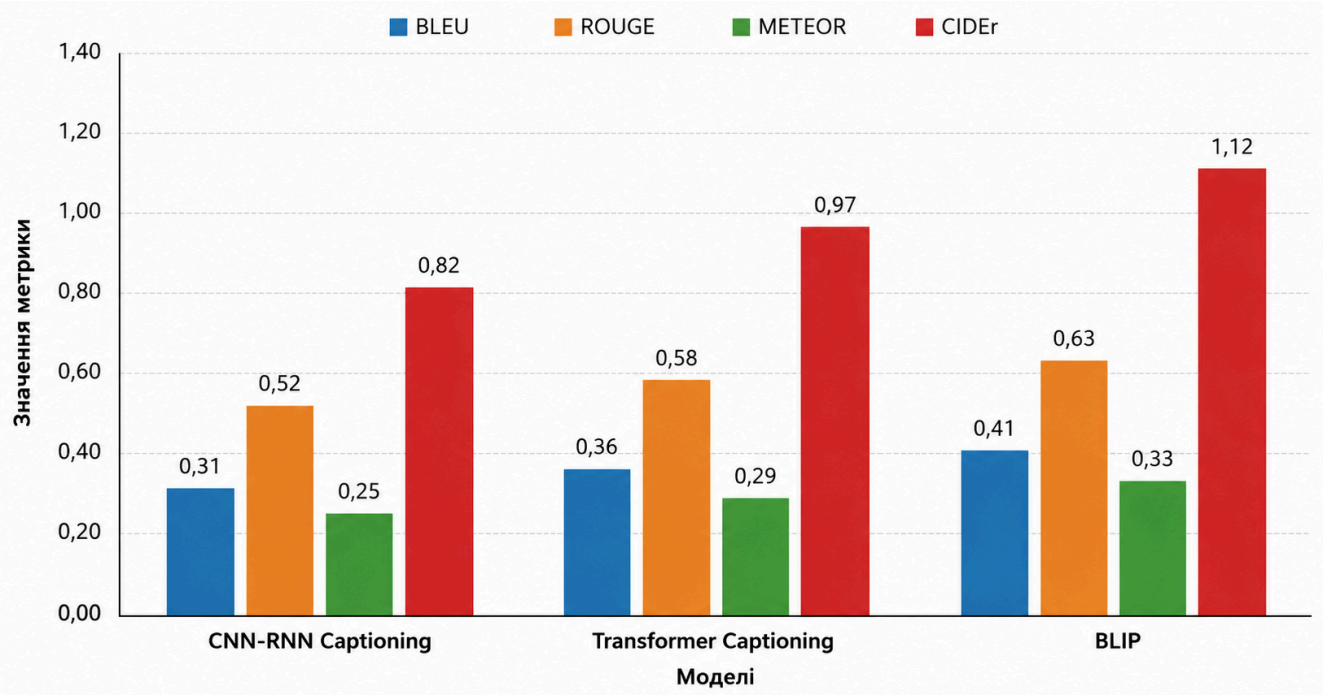

Рисунок 3.1 – Порівняння якості генерації текстових описів різних моделей

Таблиця 3.1 - Порівняння результатів різних Vision-Language моделей

Модель	BLEU	ROUGE	METEOR	CIDEr
CNN-RNN Captioning	0.31	0.52	0.25	0.82
Transformer Captioning	0.36	0.58	0.29	0.97
BLIP	0.41	0.63	0.33	1.12

Під час тестування модель BLIP продемонструвала найкращі результати за метриками BLEU та CIDEr, що свідчить про високу якість генерації описів та кращу узгодженість тексту із вмістом зображення. Крім цього, модель забезпечила стабільну генерацію описів для різних типів сцен та об’єктів, що підтверджує ефективність використання Transformer-based Vision-Language підходів у задачах image captioning.

3.3 Тестування та аналіз результатів роботи програмної системи

Після завершення реалізації програмної системи було проведено комплексне тестування основних функціональних компонентів, API взаємодії,

користувацького інтерфейсу та Vision-Language моделі. Основною метою тестування була перевірка коректності роботи системи, стабільності генерації текстових описів та оцінювання продуктивності програмного забезпечення під час обробки зображень.

Функціональне тестування підтвердило коректність реалізації основних можливостей системи, зокрема завантаження зображень, попередньої обробки даних, генерації текстових описів та відображення результатів у користувацькому інтерфейсі. Також було перевірено механізми роботи з різними моделями та обробку помилок у випадку некоректних вхідних даних.

Під час тестування API та користувацького інтерфейсу виконувалась перевірка передачі HTTP-запитів між клієнтською та серверною частинами системи. Аналіз показав стабільну взаємодію між компонентами архітектури та коректне формування JSON-відповідей після завершення генерації текстового опису. Додатково оцінювалась швидкість реакції інтерфейсу та зручність взаємодії користувача із системою.

Для аналізу продуктивності системи оцінювались час генерації текстового опису, використання GPU-ресурсів та стабільність роботи моделі під час обробки різних типів зображень. У середньому генерація одного текстового опису займала декілька секунд, що забезпечує можливість практичного використання системи у режимі реального часу.

Результати роботи Vision-Language моделі показали здатність формувати змістовні та логічно узгоджені текстові описи для більшості зображень тестової вибірки. Приклади генерації описів наведено на рисунку 3.2.

Під час аналізу результатів також були виявлені окремі помилки генерації, пов’язані зі складними сценами, великою кількістю об’єктів або недостатньою деталізацією зображень. У деяких випадках модель могла формувати занадто загальні описи або частково втрачати контекст сцени. Незважаючи на це, використання Transformer-based Vision-Language моделі забезпечило значно кращу якість генерації у порівнянні з класичними CNN-RNN підходами.

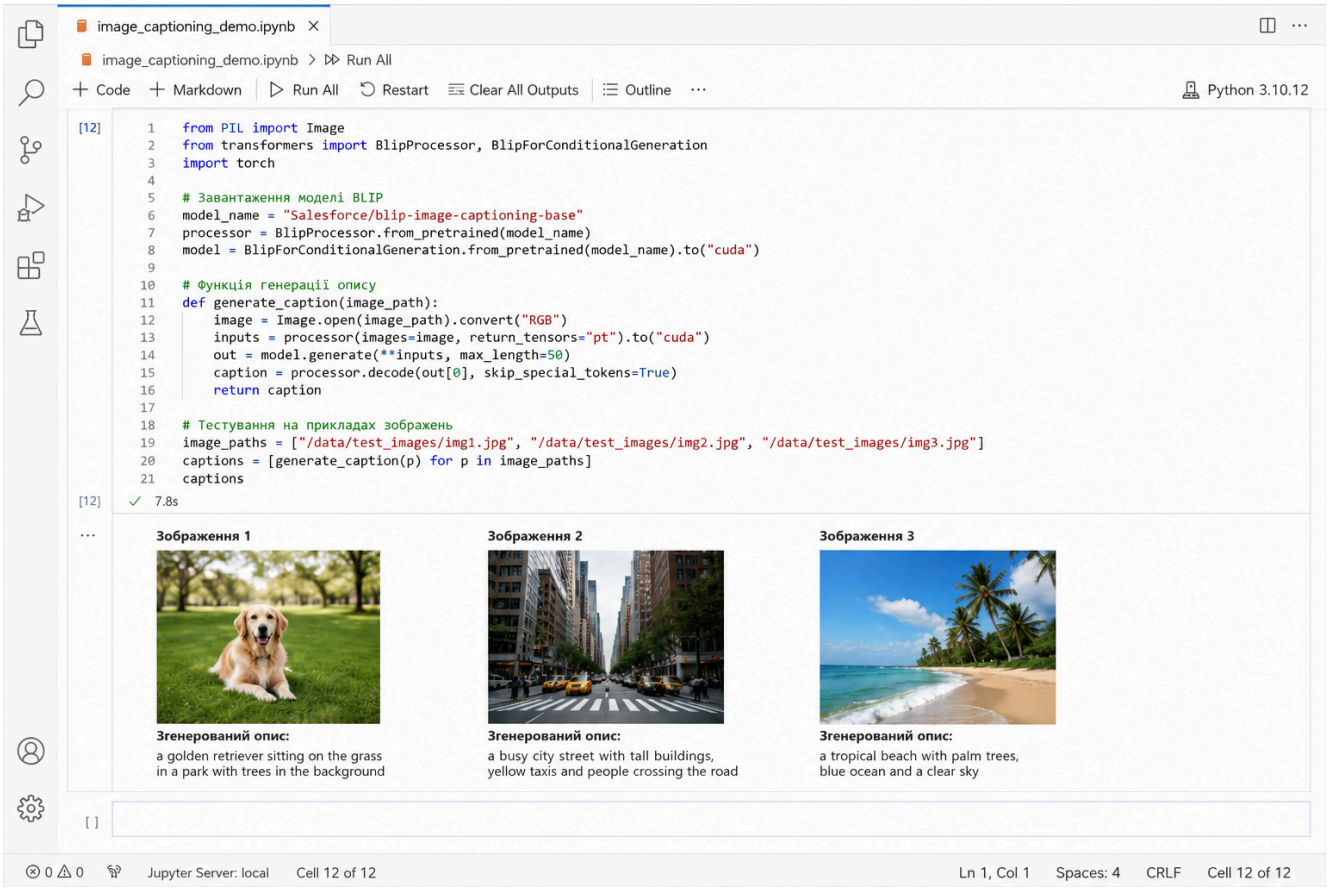

Рисунок 3.2 – Приклади генерації текстових описів зображень

Основними перевагами реалізованої системи є модульна архітектура, підтримка сучасних Vision-Language моделей, можливість масштабування та інтеграції нових моделей у межах існуючої системи. До недоліків можна віднести високу обчислювальну складність навчання Transformer-моделей та залежність якості генерації від структури й різноманітності навчального датасету.

3.4 Тестування та аналіз результатів роботи програмної системи

Подальший розвиток програмної системи може бути спрямований на розширення функціональних можливостей Vision-Language моделі, покращення якості генерації текстових описів та масштабування системи для роботи з більшими обсягами даних.

Основні перспективи розвитку системи:

- генерація текстових описів українською мовою із використанням мультимовних Transformer-моделей;
- інтеграція сучасних великих мультимодальних моделей для покращення якості та контекстності генерації описів;
- розгортання системи у хмарному середовищі для підтримки віддаленої обробки зображень та масштабування обчислювальних ресурсів;
- підтримка роботи з відеоданими та послідовностями кадрів;
- оптимізація швидкодії системи для роботи в режимі реального часу;
- розширення підтримки інших мультимодальних датасетів та моделей.

3.5 Висновки до 3 розділу

У третьому розділі було проведено тестування та оцінювання ефективності розробленої Vision-Language системи генерації текстових описів зображень. Виконано аналіз якості генерації тексту за допомогою метрик BLEU, ROUGE, METEOR та CIDEr, проведено порівняння різних моделей генерації описів та API та серверної частини системи. Отримані результати підтвердили працездатність програмного забезпечення, стабільність роботи моделі та ефективність використання Transformer-based підходів для задач image captioning, а також дозволили визначити основні переваги, обмеження та перспективи подальшого розвитку системи.

4. ОХОРОНА ПРАЦІ ТА БЕЗПЕКА В НАДЗВИЧАЙНИХ СИТУАЦІЯХ

У даному розділі розглядаються основні вимоги охорони праці та безпеки життєдіяльності, яких необхідно дотримуватися під час виконання робіт з використанням комп'ютерної техніки. Особливу увагу приділено організації робочого місця та забезпеченню безпечних умов праці відповідно до чинних нормативних документів.

4.1 Охорона праці

Під час розроблення моделі машинного навчання для генерації текстових описів зображень важливим аспектом є забезпечення належного рівня безпеки на всіх етапах життєвого циклу програмної системи. Менеджмент безпеки передбачає комплекс організаційних та технічних заходів, спрямованих на виявлення потенційних небезпек, оцінювання ризиків та впровадження засобів їх мінімізації [21].

У межах виконання кваліфікаційної роботи основна увага приділялася ризикам, пов'язаним із використанням комп'ютерної техніки, зберіганням даних, функціонуванням програмного забезпечення та взаємодією користувача із системою. Ефективний менеджмент безпеки дозволяє забезпечити стабільну роботу програмної системи, захист інформації та безпечні умови виконання робіт [21].

Одним із ключових етапів менеджменту безпеки є ідентифікація потенційних загроз. Для програмної системи генерації текстових описів зображень до основних ризиків можна віднести:

- втрату або пошкодження навчальних даних;
- збої програмного забезпечення під час навчання або тестування моделі;
- втрату результатів експериментів через відмову обладнання;

- несанкціонований доступ до даних або програмного коду;
- помилки користувача під час налаштування або використання системи;
- перебої електроживлення під час виконання обчислень;
- перевантаження обчислювальних ресурсів під час навчання моделі;
- втрату працездатності системи через програмні помилки.

Після виявлення потенційних загроз було визначено заходи щодо зниження їхнього впливу. Для запобігання втраті даних рекомендується використовувати резервне копіювання датасетів, програмного коду та результатів експериментів. Для зменшення ризику відмов програмного забезпечення доцільно виконувати регулярне тестування компонентів системи та контролювати коректність роботи алгоритмів на кожному етапі розроблення.

Важливим напрямом менеджменту безпеки є забезпечення інформаційної безпеки. Під час роботи з наборами даних та програмними компонентами рекомендується використовувати механізми автентифікації користувачів, контроль доступу до інформаційних ресурсів та засоби захисту від несанкціонованого втручання. Це дозволяє знизити ризик втрати або модифікації важливих даних.

Особливу увагу необхідно приділяти забезпеченню безперервності роботи системи. Для цього рекомендується використовувати стабільні джерела електроживлення, регулярно зберігати результати навчання моделей та застосовувати системи контролю версій для програмного коду. Використання таких заходів дозволяє мінімізувати наслідки технічних збоїв та підвищити надійність програмної системи.

У процесі виконання роботи також враховувалися питання безпеки праці користувача під час роботи з комп’ютерною технікою. Організація робочого місця здійснювалася відповідно до вимог Закону України «Про охорону праці» [21], НПАОП 0.00-7.15-18 [22] та ДСанПіН 3.3.2.007-98 [23]. Дотримання зазначених вимог дозволяє зменшити вплив шкідливих факторів виробничого середовища та забезпечити безпечне виконання робіт.

Основні заходи менеджменту безпеки, які доцільно використовувати під час експлуатації системи:

- резервне копіювання даних та результатів навчання моделей;
- використання систем контролю версій програмного коду;
- контроль доступу до програмних та інформаційних ресурсів;
- регулярне тестування програмного забезпечення;
- використання антивірусного програмного забезпечення;
- моніторинг використання обчислювальних ресурсів;
- забезпечення стабільного електроживлення обладнання;
- дотримання вимог охорони праці та ергономіки робочого місця;
- документування змін програмного забезпечення та результатів експериментів;
- проведення періодичної оцінки ризиків під час супроводу системи.

Таким чином, менеджмент безпеки є важливою складовою процесу розроблення та експлуатації програмної системи генерації текстових описів зображень. Впровадження заходів щодо управління ризиками дозволяє підвищити надійність функціонування системи, забезпечити захист інформації та створити безпечні умови для роботи користувачів. Запропоновані заходи рекомендується використовувати під час подальшого розвитку та впровадження програмної системи [21–23].

4.2 Заходи, що покращують умови праці оператора

Під час розроблення моделі машинного навчання для генерації текстових описів зображень значна частина робіт виконувалася з використанням персонального комп’ютера, тому особлива увага приділялася забезпеченню безпечних та комфортних умов праці оператора. Організація робочого місця здійснювалася відповідно до вимог Закону України «Про охорону праці», який встановлює обов’язок створення безпечних і нешкідливих умов праці для працівників [21].

Важливим елементом забезпечення безпечних умов праці є дотримання вимог електробезпеки під час експлуатації комп’ютерної техніки. Усі технічні засоби повинні бути підключені до справної електромережі, а кабелі живлення та комунікаційні лінії мають бути розташовані таким чином, щоб виключити можливість їх механічного пошкодження або випадкового контакту з працівником. Дотримання цих вимог знижує ризик виникнення аварійних ситуацій та забезпечує безпечну експлуатацію обладнання [21, 22].

Робоче місце було організоване з урахуванням вимог НПАОП 0.00-7.15-18 щодо безпечної роботи з екранними пристроями [22]. Під час виконання роботи забезпечувалася достатня відстань між користувачем і монітором, використовувалося обладнання з якісним відображенням інформації, а також підтримувався раціональний режим праці та відпочинку. Такі заходи дозволяють зменшити навантаження на органи зору та знизити ризик виникнення професійної втоми.

Для створення комфортного виробничого середовища враховувалися вимоги ДСанПіН 3.3.2.007-98 щодо умов праці користувачів комп’ютерної техніки [3]. Було забезпечено належний рівень освітлення робочої зони, підтримання сприятливого мікроклімату приміщення та мінімізацію впливу сторонніх шумів. Дотримання зазначених вимог позитивно впливає на працездатність оператора та якість виконання робіт.

Особлива увага приділялася ергономічній організації робочого місця відповідно до положень ДСТУ ISO 9241-5:2004 [24]. Монітор було розташовано на оптимальній висоті відносно рівня очей користувача, робочий стіл забезпечував достатній простір для розміщення обладнання, а крісло дозволяло підтримувати правильне положення тіла під час роботи. Такі рішення сприяють зменшенню навантаження на хребет, шийний відділ та верхні кінцівки.

З метою покращення умов праці оператора під час розроблення та тестування програмної системи були реалізовані такі заходи:

- забезпечено достатній рівень природного та штучного освітлення робочої зони;

- виключено появу відблисків та засвіток на поверхні монітора;
- організовано правильне розташування монітора, клавіатури та маніпулятора типу «миша»;
- забезпечено підтримання нормативних параметрів температури та вологості повітря;
- використано ергономічне крісло з можливістю регулювання висоти та положення спинки;
- забезпечено достатній простір для ніг та вільного розміщення обладнання на робочому столі;
- організовано регулярні перерви під час тривалої роботи за комп’ютером;
- передбачено виконання вправ для зниження зорової втоми;
- забезпечено безпечне підключення комп’ютерного обладнання до електромережі;
- мінімізовано вплив шуму та інших факторів, що можуть знижувати концентрацію уваги оператора.

Під час виконання роботи також враховувалися рекомендації щодо організації праці користувачів комп’ютерної техніки, наведені у спеціалізованій літературі [25]. Для зниження психоемоційного навантаження рекомендується періодично змінювати вид діяльності, виконувати вправи для очей та робити короткочасні перерви протягом робочого дня. Особливо актуальними такі заходи є під час тривалого навчання моделей машинного навчання, аналізу результатів експериментів та роботи з великими обсягами даних.

Окрім фізичних факторів виробничого середовища, важливе значення мають психофізіологічні умови праці оператора. Під час розроблення та тестування моделей машинного навчання оператор виконує значний обсяг аналітичної роботи, пов’язаної з налаштуванням параметрів моделей, контролем процесу навчання та оцінюванням отриманих результатів. Така діяльність характеризується високою концентрацією уваги та значним інтелектуальним навантаженням. Для запобігання перевтомі рекомендується раціонально

планувати робочий час, чергувати різні види діяльності та дотримуватися встановлених режимів праці та відпочинку відповідно до вимог нормативних документів [22, 25].

Отже, під час виконання кваліфікаційної роботи були враховані вимоги законодавчих, нормативних та методичних документів у сфері охорони праці [21–25]. Запропоновані заходи доцільно використовувати і під час подальшої експлуатації та супроводу програмної системи, оскільки вони сприяють підвищенню продуктивності праці оператора, збереженню його працездатності та зменшенню впливу несприятливих виробничих факторів.

ВИСНОВКИ

У ході дослідження було проаналізовано сучасні набори даних для задачі генерації текстових описів зображень, зокрема MS COCO Captions, Flickr8k та Flickr30k. Встановлено, що датасет MS COCO Captions є найбільш придатним для навчання Vision-Language моделей завдяки значному обсягу даних, різноманітності сцен та наявності декількох текстових описів для кожного зображення. Перед початком навчання було виконано попередню обробку даних, яка включала підготовку зображень, очищення текстових анотацій та формування навчальної, валідаційної та тестової вибірок.

У процесі роботи було проведено аналіз сучасних підходів до генерації текстових описів зображень. Розглянуто архітектури CNN-RNN, Transformer-підходи та мультимодальні Vision-Language моделі. Результати аналізу показали, що класичні CNN-RNN архітектури поступаються сучасним Transformer-моделям за якістю генерації тексту та здатністю враховувати контекст сцени. Найбільш перспективними для задачі image captioning виявилися мультимодальні моделі сімейства BLIP, які поєднують високу точність генерації та ефективне узгодження візуальних і текстових представлень.

На основі проведеного аналізу було розроблено програмну систему для автоматичної генерації текстових описів зображень. Для реалізації моделі використано сучасні засоби машинного навчання та бібліотеки комп’ютерного зору. Побудована архітектура забезпечує обробку вхідного зображення, виділення візуальних ознак та формування текстового опису природною мовою. Розроблене програмне рішення має модульну структуру та може бути адаптоване для використання з іншими наборами даних або Vision-Language моделями.

Для оцінювання якості роботи моделі було використано спеціалізовані метрики BLEU, ROUGE, METEOR та CIDEr, які широко застосовуються у задачах image captioning. Отримані результати підтвердили здатність моделі формувати змістовні та граматично коректні описи зображень. Проведене тестування показало, що використання сучасних Vision-Language підходів дозволяє досягти

високої узгодженості між візуальним вмістом зображення та сформованим текстовим описом.

Важливим етапом дослідження стала оптимізація параметрів моделі та аналіз їх впливу на якість генерації описів. Було досліджено вплив розміру навчальної вибірки, кількості епох, швидкості навчання та інших параметрів на значення метрик BLEU, ROUGE, METEOR і CIDEr. Отримані результати показали, що коректне налаштування гіперпараметрів дозволяє підвищити якість генерації тексту та забезпечити більш стабільну роботу моделі на нових даних.

Практична цінність розробленої системи полягає у можливості автоматичного формування текстових описів зображень для різних предметних областей. Запропоноване рішення може бути використане в системах доступності для людей із порушеннями зору, сервісах автоматичного аналізу мультимедійного контенту, пошукових системах, цифрових архівах та інтелектуальних вебплатформах.

У результаті виконання роботи було досягнуто поставленої мети – розроблено модель машинного навчання для генерації текстових описів зображень на основі Vision-Language підходів. Проведені дослідження підтвердили ефективність використання сучасних мультимодальних моделей для поєднання візуальної та текстової інформації. Отримані результати продемонстрували можливість формування змістовних описів зображень із високим рівнем відповідності їх вмісту.

Подальший розвиток роботи може бути пов’язаний із використанням більш потужних мультимодальних моделей, зокрема нових версій BLIP, Florence або інших Vision-Language архітектур. Перспективними напрямами також є підтримка української мови, розширення наборів даних, використання великих мовних моделей для покращення якості тексту та розгортання системи у хмарному середовищі для обробки великих обсягів зображень. Це дозволить підвищити якість генерації описів та розширити сферу практичного застосування розробленого програмного забезпечення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Михалик Д. М., Цуприк Г. Б., Бревус В. М. Методичні вказівки до виконання кваліфікаційної роботи бакалавра для здобувачів першого (бакалаврського) рівня вищої освіти за освітньо-професійною програмою «Інженерія програмного забезпечення» спеціальності 121 – «Інженерія програмного забезпечення» всіх форм навчання. Тернопіль : ТНТУ ім. І. Пулюя, 2024. 45 с.

2. Бондаренко М. Ф., Каторгін І. В., Моторін Р. М. Штучний інтелект та машинне навчання : навчальний посібник. Київ : КПІ ім. Ігоря Сікорського, 2023. 312 с.

3. Лупенко С. А., Пасічник В. В., Яцишин В. С. Аналіз даних та інтелектуальні системи : навчальний посібник. Тернопіль : ТНТУ ім. І. Пулюя, 2023. 286 с.

4. Goodfellow I., Bengio Y., Courville A. Deep Learning. Cambridge : MIT Press, 2016. 800 p.

5. Bishop C. M. Pattern Recognition and Machine Learning. New York : Springer, 2006. 738 p.

6. Géron A. Hands-On Machine Learning with Scikit-Learn, Keras and TensorFlow. 3rd ed. Sebastopol : O'Reilly Media, 2022. 851 p.

7. Raschka S., Liu Y., Mirjalili V. Machine Learning with PyTorch and Scikit-Learn. Birmingham : Packt Publishing, 2022. 770 p.

8. Szeliski R. Computer Vision: Algorithms and Applications. 2nd ed. Cham : Springer, 2022. 1118 p.

9. Prince S. J. D. Computer Vision: Models, Learning, and Inference. Cambridge : Cambridge University Press, 2012. 598 p.

10. Vaswani A. et al. Attention Is All You Need // Advances in Neural Information Processing Systems. 2017. Vol. 30. P. 5998–6008.

11. Dosovitskiy A. et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale // International Conference on Learning Representations (ICLR). 2021.

12. Radford A. et al. Learning Transferable Visual Models From Natural Language Supervision // Proceedings of the International Conference on Machine Learning. 2021. Vol. 139. P. 8748–8763.

13. Li J., Li D., Savarese S., Hoi S. C. H. BLIP: Bootstrapping Language-Image Pre-training for Unified Vision-Language Understanding and Generation // Proceedings of the International Conference on Machine Learning. 2022. P. 12888–12900.

14. Li J. et al. BLIP-2: Bootstrapping Language-Image Pre-training with Frozen Image Encoders and Large Language Models // International Conference on Machine Learning. 2023. P. 19730–19742.

15. Xiao D. et al. Florence: A New Foundation Model for Computer Vision // arXiv preprint arXiv:2111.11432. 2021.

16. Lin T.-Y. et al. Microsoft COCO: Common Objects in Context // European Conference on Computer Vision. Cham : Springer, 2014. P. 740–755.

17. Young P., Lai A., Hodosh M., Hockenmaier J. From Image Descriptions to Visual Denotations: New Similarity Metrics for Semantic Inference over Event Descriptions // Transactions of the Association for Computational Linguistics. 2014. Vol. 2. P. 67–78.

18. Plummer B. A. et al. Flickr30k Entities: Collecting Region-to-Phrase Correspondences for Richer Image-to-Sentence Models // International Journal of Computer Vision. 2017. Vol. 123. No. 1. P. 74–93.

19. Papineni K., Roukos S., Ward T., Zhu W.-J. BLEU: A Method for Automatic Evaluation of Machine Translation // Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics. 2002. P. 311–318.

20. Vedantam R., Zitnick C. L., Parikh D. CIDEr: Consensus-Based Image Description Evaluation // Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015. P. 4566–4575.

21. Про охорону праці : Закон України від 14.10.1992 № 2694-XII. // База даних «Законодавство України» / Верховна Рада України.

22. Про затвердження Вимог щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями : Наказ Міністерства соціальної політики України від 14.02.2018 № 207 (НПАОП 0.00-7.15-18). // База даних «Законодавство України» / Верховна Рада України.

23. ДСанПіН 3.3.2.007-98. Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин.

24. ДСТУ ISO 9241-5:2004. Ергономічні вимоги до роботи з відеотерміналами в офісі. Частина 5. Вимоги до компонування робочого місця та до робочої пози.

25. Жидецький В.Ц. Охорона праці користувачів комп'ютерів : підручник. – Львів : Афіша, 2020. – 176 с.

ДОДАТКИ

ДОДАТОК А – Рисунок основної діаграми послідовності

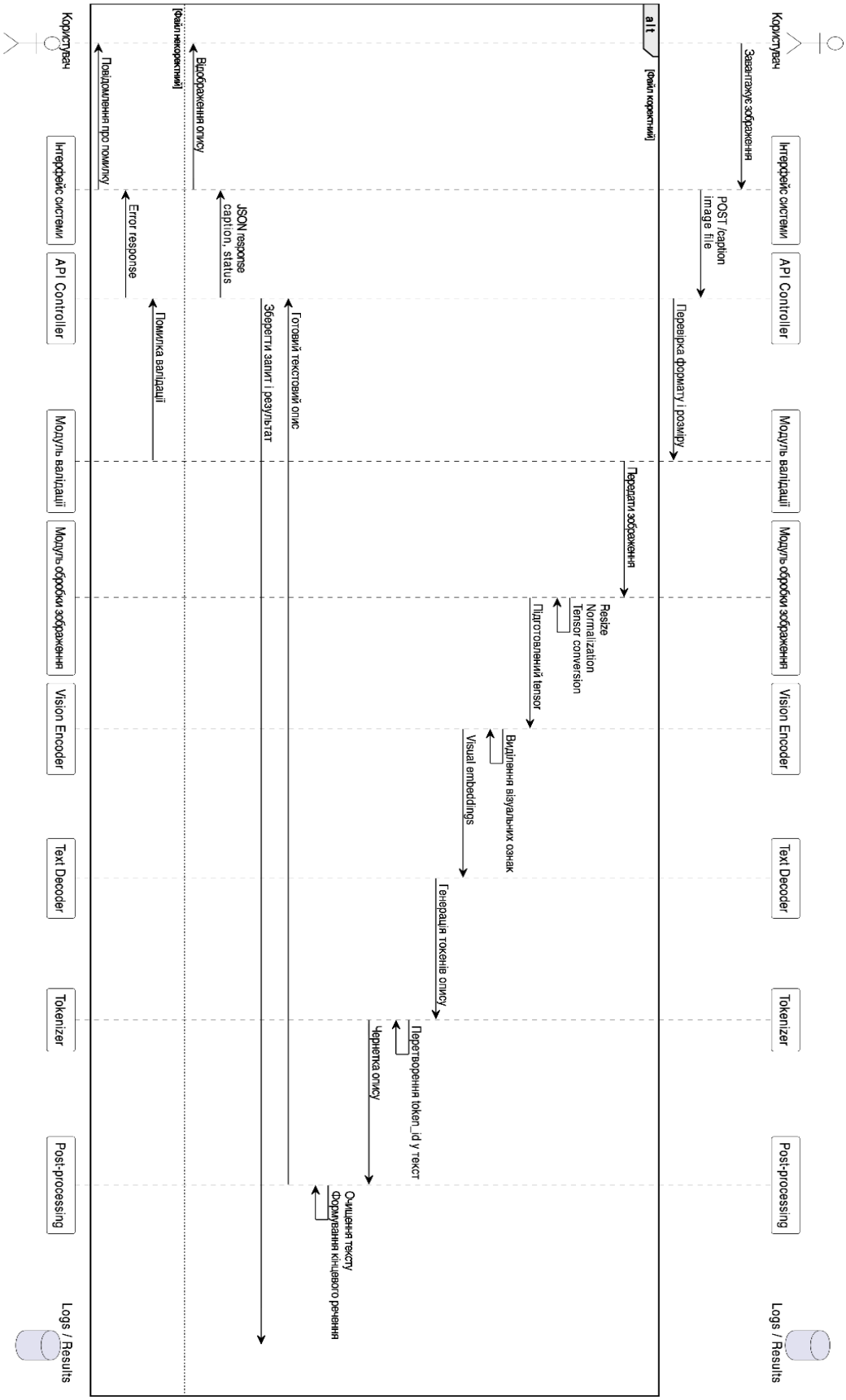

Рисунок А.1 - Діаграма послідовності

ДОДАТОК Б – Теза конференції

УДК 621.326
Мигаль З. – ст. гр. СП-42
Тернопільський національний технічний університет імені Івана Пулюя

РОЗРОБКА МОДЕЛІ МАШИННОГО НАВЧАННЯ ДЛЯ ГЕНЕРАЦІЇ ТЕКСТОВИХ ОПИСІВ ЗОБРАЖЕНЬ НА ОСНОВІ VISION-LANGUAGE ПІДХОДІВ

Науковий керівник: к.ф.-м.н., доцент Цебрій О. Р.

Myhal Z.
Ternopil Ivan Puluj National Technical University

DEVELOPMENT OF A MACHINE LEARNING MODEL FOR GENERATING IMAGE CAPTIONS BASED ON VISION-LANGUAGE APPROACHES

Supervisor: PhD in Physics and Mathematics, Associate Professor Tsebrii O.

Ключові слова: генерація тексту, комп'ютерний зір, vision-language
Keywords: text generation, computer vision, vision-language

Задача автоматичної генерації текстових описів зображень є важливим напрямом сучасних досліджень у сфері штучного інтелекту, оскільки вона поєднує методи комп'ютерного зору та обробки природної мови. Такі системи знаходять застосування у допоміжних технологіях для людей з порушеннями зору, системах пошуку зображень, автоматичному тегуванні контенту та цифрових асистентах. Основною складністю задачі є необхідність коректного поєднання візуальної інформації з мовними конструкціями, що потребує глибокого розуміння як зображення, так і контексту.

Сучасні підходи базуються на використанні глибоких нейронних мереж, зокрема згорткових нейронних мереж для витягування ознак із зображень та трансформерних моделей для генерації тексту. Архітектури типу encoder-decoder дозволяють перетворювати візуальні ознаки у послідовності слів, формуючи осмислені описи. Використання attention-механізмів забезпечує можливість фокусування на окремих ділянках зображення під час генерації кожного слова, що підвищує якість та релевантність описів.

Окрім класичних підходів, активно розвиваються vision-language моделі, такі як CLIP, BLIP та їх похідні, які навчаються на великих мультимодальних датасетах і здатні ефективно узгоджувати текстові та візуальні представлення. Такі моделі демонструють високу узагальнювальну здатність і можуть застосовуватись у широкому спектрі задач без значного донавчання.

Важливим етапом є підготовка даних, яка включає формування пар «зображення–опис», очищення тексту та нормалізацію зображень. Якість датасету безпосередньо впливає на результати моделі, оскільки некоректні або неповні описи можуть призводити до помилок у генерації. Для оцінювання якості використовуються метрики, такі як BLEU, METEOR, CIDEr та ROUGE, що дозволяє комплексно оцінити відповідність згенерованого тексту реальному опису [1].

Важливим є врахування контексту та семантичної узгодженості описів. Модель повинна не лише ідентифікувати об'єкти на зображенні, але й правильно описувати їх

взаємодію, просторове розташування та дії. Для цього використовуються механізми попереднього навчання на великих корпусах даних, що дозволяє покращити мовну складову системи.

Додатково важливим аспектом є вибір способу представлення візуальних ознак, які передаються у мовну модель. Сучасні підходи використовують як глобальні представлення зображення, так і локальні ознаки окремих регіонів, що дозволяє більш точно описувати складні сцени. Використання регіональних фіч або patch-представлень у трансформерних моделях забезпечує кращу деталізацію описів та підвищує здатність моделі враховувати дрібні об'єкти та їх взаємозв'язки.

Особливу роль відіграє процес узгодження візуального та текстового простору ознак. Для цього застосовуються контрастивні методи навчання, які дозволяють зблизити відповідні пари «зображення–текст» та віддалити невідповідні. Такий підхід підвищує якість семантичного розуміння та дозволяє моделі краще узагальнювати нові дані. Крім того, це відкриває можливості для використання моделі у суміжних задачах, таких як пошук зображень за текстовим запитом або навпаки.

Також важливим є врахування якості та різноманітності текстових описів, оскільки одна сцена може мати декілька коректних варіантів опису. Для вирішення цієї проблеми застосовуються методи генерації з використанням стохастичних підходів, таких як beam search або sampling, що дозволяє отримувати більш природні та варіативні результати. Це підвищує гнучкість системи та робить її більш придатною для реальних застосувань, де важлива не лише точність, але й природність сформованого тексту.

Окрему увагу приділяють оптимізації моделей та їх впровадженню у практичні системи. Методи стиснення моделей, такі як квантизація та дистиляція знань, дозволяють зменшити обчислювальні витрати та забезпечити роботу в реальному часі. Це відкриває можливості використання таких систем у мобільних додатках та веб-сервісах [2].

Таким чином, поєднання методів комп'ютерного зору та обробки природної мови у рамках vision-language підходів формує цілісну технологічну основу для побудови ефективних систем генерації текстових описів зображень. Важливим є не лише використання окремих моделей, а їх узгоджена інтеграція у єдину архітектуру, де візуальні ознаки коректно трансформуються у мовні представлення з урахуванням контексту та семантики. Завдяки використанню попередньо навчених мультимодальних моделей та великих датасетів досягається висока якість узагальнення, що дозволяє системам працювати з різними типами зображень та сценаріями застосування без значного донавчання.

У результаті формується інтелектуальна система, яка може бути інтегрована у веб-сервіси, мобільні додатки та інші цифрові платформи, забезпечуючи більш природну та ефективну взаємодію людини з інформаційними системами. Це сприяє розширенню можливостей доступу до інформації, покращенню якості цифрових сервісів та створенню нових підходів до обробки й інтерпретації мультимедійних даних.

Література:

1. Radford, A. et al. (2021). Learning Transferable Visual Models From Natural Language Supervision. — ICML.

2. Li, J., Li, D., Savarese, S., Hoi, S. (2022). BLIP: Bootstrapping Language-Image Pre-training. — ICML.

3. Alayrac, J.-B., Donahue, J., Luc, P., et al. (2022). Flamingo: a Visual Language Model for Few-Shot Learning. — NeurIPS.