Міністерство освіти і науки України
Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)
Кафедра Штучного інтелекту
(повна назва)

КВАЛІФІКАЦІЙНА РОБОТА
Пояснювальна записка

рівень вищої освіти другий (магістерський)

Дослідження нейромережевих методів зіставлення зображень та їх текстових анотацій
(тема)

Виконав:
студент 2 курсу, групи СШІм-20-2
Потапов Д.С.
(прізвище, ініціали)

Спеціальність 122 Комп'ютерні науки
(код і повна назва спеціальності)
Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)
Освітня програма Системи штучного інтелекту
(повна назва спеціалізації)
Керівник проф. Рябова Н.В.
(посада, прізвище, ініціали)

Допускається до захисту

Зав. кафедри _____ В.О. Філатов
(підпис) (прізвище, ініціали)

2022

Харківський національний університет радіоелектроніки

Факультет Комп'ютерних наук
(повна назва)

Кафедра Штучного інтелекту
(повна назва)

Рівень вищої освіти другий (магістерський)

Спеціальність 122 Комп'ютерні науки
(код і повна назва)

Тип програми освітньо-наукова
(освітньо-професійна або освітньо-наукова)

Освітня програма Системи штучного інтелекту (СШІ)
(повна назва)

ЗАТВЕРДЖУЮ:
Зав. кафедри _____
(підпис)
«_____»_____20____р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

студентові Потапову Дмитру Станіславовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Дослідження нейромережевих методів зіставлення зображень та їх текстових анотацій

затверджена наказом університету від 24 березня 2022 р. № 414Ст
2. Термін подання студентом роботи до екзаменаційної комісії 12 травня 2022 р.
3. Вихідні дані до роботи Науково-технічна публікації, дані Інтернет-джерел та відомих наукових проектів щодо розробки та дослідження різних нейромережевих методів зіставлення зображень з їх текстовою анотацією

4. Перелік питань, що потрібно опрацювати в роботі
1) Аналіз предметної області дослідження
2) Огляд існуючих нейромережевих методів для перетворення текстового опису в зображення
3) Опис уважно змагальної генеративної мережі для вирішення задачі перетворення
4) Програмна реалізація веб-додатку для демонстрації отриманих результатів

5. Перелік графічного матеріалу із зазначенням креслеників, схем, плакатів, комп'ютерних ілюстрацій (п.5 включається до завдання за рішенням випускової кафедри)

Рисунок 2.1 – Зразок формування тексту, Рисунок 2.2 – Текстова умовно-згорткова архітектруа GAN, Рисунок 2.3 – Існуючі датасети, Рисунок 2.4 – Приклад перетворення з T2I методом, Рисунок 2.5 –Принцип роботи архітектури T2I, Рисунок 2.6 – Архітектура GAN-INT-CLS, Рисунок 2.7 – Архітектура StackGAN та StackGAN++, Рисунок 2.8 – Архітектура AttnGAN, Рисунок 2.9 – Архітектура SD-GAN, Рисунок 2.10 – Архітектура MirrorGAN, Рисунок 2.11 – Компоненти фреймворку MirrorGAN, Рисунок 2.12 – Принцип роботи STEM, GLAM, STREAM, Рисунок 2.13 – Архітектура StyleGAN, Рисунок 2.14 – Приклад роботи MSG-GAN, Рисунок 2.15 – Архітектури MSG-GAN, Рисунок 2.16 – Приклад обєктно-керованого синтезу текста, Рисунок 2.17 – Принцип роботи Obj-GAN, Рисунок 2.18 – Архітектура StoryGAN, Рисунок 2.19 – Структура історії дискримінатора, Рисунок 2.20 – Результат генерації гляхом зміни імен персонажів, Рисунок 2.21 – Приклад архітектури DCGAN для генерації зображень, Рисунок 3.1 – Загальна архітектура AttnGAN, Рисунок 3.2 – Приклад результатів запропонованого AttnGAN, Рисунок 4.1 – Інтерфейс веб-додатку, Рисунок 4.2 – Перша згенерована фотографія, Рисунок 4.3 – Друга згенерована фотографія, Рисунок 4.4 – Третя згенерована фотографія, Рисунок 4.5 – Статистика датасетів, Рисунок 4.6 – Початкові оцінки та показники точності R, Рисунок 4.7 – Найкращий початковий бал, Рисунок 4.8 – Проміжні результати, Рисунок 4.9 – Приклади результатів, Рисунок 4.10 – Нові зображення

6. Консультанти розділів роботи (п.6 включається до завдання за наявності консультантів згідно з наказом, зазначеним у п.1)

Найменування розділу	Консультант (посада, прізвище, ім'я, по батькові)	Позначка консультанта про виконання розділу	
		підпис	дата
Основна частина	проф. Рябова Н.В.		

КАЛЕНДАРНИЙ ПЛАН

№	Назва етапів роботи	Терміни виконання етапів роботи	Примітка
1	Отримання завдання	28.03.2022	виконано
2	Аналіз предметної області і постановка задачі	29.03.2022	виконано
3	Аналіз методів щодо вирішення задачі	31.03.2022	виконано
4	Розробка вимог до програмної системи	02.04.2022	виконано
5	Розробка системи	04.04.2022	виконано
6	Програмна реалізація	04.04.2022	виконано
7	Аналіз результатів	12.04.2022	виконано
8	Оформлення пояснювальної записки	15.04.2022	виконано
9	Попередній захист	12.05.2022	виконано
10	Захист перед ЕК	16.05.2022	виконано

Дата видачі завдання 28 березня 2022 р.

Студент _____
(підпис)

Керівник роботи _____ проф. Рябова Н.В.
(підпис) (посада, прізвище, ініціали)

РЕФЕРАТ

Пояснювальна записка: 79 с., 33 рис., 1 дод., 44 джерел.

ГЕНЕРАЦІЯ ТЕКСТУ В ЗОБРАЖЕННЯ, МЕТОДИ ПЕРЕТВОРЕННЯ, НЕЙРОННІ МЕРЕЖІ, ГЕНЕРАТИВНО-ЗМАГАЛЬНА МЕРЕЖА, GAN, T2I

Об'єкт дослідження – задача обробки та зіставлення даних різних модальностей (текст на зображення).

Предмет дослідження – архітектури нейронних мереж для трансформації текстової інформації у зображення.

Мета роботи – обґрунтування, вибір та дослідження архітектур та методів генеративного моделювання з метою виявлення найбільш творчих та ефективних у навчанні моделей та розробка веб-додатку для демонстрації отриманих результатів.

Методи дослідження – методи машинного та глибинного навчання, аналіз основних властивостей та можливих архітектур генеративних змагальних мереж, порівняльний аналіз експериментів та отриманих результатів.

Проведено теоретичний аналіз різних архітектур змагальних генеративних нейронних мереж та методів перетворення тексту в зображення.

В результаті проведення досліджень вирішено задачу перетворення тексту в зображення за допомогою уважно генеративної змагальної нейронної мережі AttnGAN. Отримані результати мають змогу використовуватися в різних напрямках, наприклад, крос-модальний пошук інформації, редагування фотографій та автоматизований дизайн.

РЕФЕРАТ

Пояснительная записка: 79 с., 33 рис., 1 прил., 44 источников.

ГЕНЕРАЦИЯ ТЕКСТА В ИЗОБРАЖЕНИЕ, МЕТОДЫ ПРЕОБРАЗОВАНИЯ, НЕЙРОННЫЕ СЕТИ, ГЕНЕРАТИВНО-СОСТЯЗАТЕЛЬНАЯ СЕТЬ, GAN, T2I

Объект исследования – задача обработки и сопоставления данных разных модальностей (текст на изображение).

Предмет исследования – архитектура нейронных сетей для трансформации текстовой информации в изображение.

Цель работы – обоснование, выбор и исследование архитектур и методов генеративного моделирования с целью выявления наиболее творческих и эффективных в обучении моделей и разработка веб-приложения для демонстрации полученных результатов.

Методы исследования – методы машинного и глубинного обучения, анализ основных свойств и возможных архитектур генеративных состязательных сетей, сравнительный анализ экспериментов и полученных результатов.

Проведен теоретический анализ различных архитектур соревновательных генеративных нейронных сетей и методов преобразования текста в изображение.

В результате проведения исследований решена задача преобразования текста в изображение с помощью внимательно генеративной соревновательной нейронной сети AttnGAN. Полученные результаты могут использоваться по разным направлениям, например, кросс-модальный поиск информации, редактирование фотографий и автоматизированный дизайн.

ABSTRACT

Explanatory note: 79 p., 33 fig., 1 ann., 44 sources.

TEXT GENERATION INTO IMAGE, METHODS OF TRANSFORMATION, NEURAL NETWORKS, GENERATIVE COMPETITIVE NETWORK, GAN, T2I

The object of research – the task of processing and comparing data of different modalities (text to image).

The subject of research is the architecture of neural networks for the transformation of textual information into images.

The purpose of the work is to substantiate, select and study architectures and methods of generative modeling in order to identify the most creative and effective in learning models and develop a web application to demonstrate the results.

Research methods – methods of machine and deep learning, analysis of the basic properties and possible architectures of generative competitive networks, comparative analysis of experiments and results.

Theoretical analysis of different architectures of competitive generative neural networks and methods of text-to-image conversion is carried out.

As a result of research, the problem of converting text into images using a carefully generative competitive neural network AttnGAN was solved. The results can be used in various areas, such as cross-modal information retrieval, photo editing and automated design.

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ І ТЕРМІНІВ

ШІ – штучний інтелект;

AttnGAN – attentional generative adversarial network – генеративна змагальна мережа з механізмом уваги;

BigGAN – big generative adversarial network – велика генеративна змагальна мережа;

CNN-RNN – convolutional neural network – згорткова нейронна мережа;

ControlGAN – controllable generative adversarial network – керована генеративна змагальна мережа;

CycleGAN – cycle-consistent generative adversarial network – узгоджена з циклом генеративна змагальна мережа;

DAMSM – deep attentional multimodal similarity model – модель мультимодальної подібності глибокої уваги;

FusedGAN – fused generative adversarial network – зрощена генеративна змагальна мережа;

GAN – generative adversarial network – генеративна змагальна мережа;

GLAM – global-local attentive module – глобально-локальний модуль уваги;

HDGAN – hierarchically nested generative adversarial network – ієрархічно вкладена генеративна змагальна мережа;

LSTM – long short-term memory – довга короткочасна пам’ять;

MirrorGAN – generative adversarial network by redescription – генеративна змагальна мережа шляхом переопису;

MLP – multilayer perceptron – багатошаровий персептрон;

MSG-GAN – multi-scale gradient generative adversarial network – багатомасштабна градієнтна генеративна змагальна мережа;

ObjGAN – object-driven attentive generative adversarial network – об’єктно-керована уважна генеративна змагальна мережа;

ProGAN – progressive generative adversarial network – прогресивна генеративна змагальна мережа;

RNN – recurrent neural network – рекурентна нейронна мережа;

SD-GAN – semantics disentangling generative adversarial network – семантика, яка розкриває генеративну змагальну мережу;

StackGAN – stacked generative adversarial network – стекована генеративна змагальна мережа;

STEM – semantic text embedding module – модуль вбудовування семантичного тексту;

StoryGAN – generative adversarial network for story visualization – генеративна змагальна мережа для візуалізації історії;

STREAM – semantic text regeneration alignment module – модуль вирівнювання семантичної регенерації тексту;

StyleGAN – style generative adversarial network – стильна генеративна змагальна мережа;

TAC-GAN – text conditional auxiliary classifier of a generative adversarial network – умовний допоміжний класифікатор генеративної змагальної мережі;

TTI – text-to-image – текст-в-зображення;

T2I – text2image – текст в зображення;

VQA-GAN – visual question answering generative adversarial network – генеративне змагальне візуальне запитання відповідно змагальної мережі.

ВСТУП

В останні роки, з появою штучного інтелекту та глибокого навчання, обробка природної мови та комп’ютерний зір стали популярними областями досліджень. Текст до зображення як основна проблема в цій галузі також привернула увагу та дослідження багатьох вчених. Перетворення тексту на зображення – це створення реалістичного зображення, що відповідає заданому текстовому опису, що вимагає обробки нечіткої та неповної інформації в описах природною мовою [1]. Перетворення тексту на зображення стимулює розвиток мультимодального навчання та крос-модальної генерації та демонструє великий потенціал у таких додатках, як крос-модальний пошук інформації, редагування фотографій та автоматизований дизайн. За останні кілька років було проведено багато досліджень і розробок щодо створення моделей штучного інтелекту, які можуть створювати зображення з заданих текстових підказок. Це можна вважати особистим художником, який намагається створити твір мистецтва [2], дотримуючись усіх самих слів вашої інструкції.

Генерація зображення із текстового опису дві мети: візуальний реалізм та смислова узгодженість [3]. Незважаючи на значний прогрес у створенні високоякісних та візуально реалістичних зображень з використанням генеративних змагальних мереж, що гарантують семантичну узгодженість між текстовим описом та візуальним контентом, залишається дуже складним завданням. Також, ми можемо сказати, що генерація зображень вирішує два важливі завдання, які не під силу вирішити пошуковику, по-перше, вона дозволяє врахувати точний опис бажаного, а по-друге, програма створює унікальні зображення, які раніше не існували. Їх можна використовувати для фото-ілюстрацій статей, у копірайтингу та в рекламі.

У світі щоденно з'являються мільярди нових фотографій. Тому

класифікувати та організувати їх таким чином, щоб пошук конкретної групи чи унікального зображення не вимагав багато часу та зусиль – завдання досить складне.

Опис картинок за допомогою нейромереж полегшує завдання пошуку та видачі релевантних результатів у пошукових системах за запитом користувача природною мовою. Можна автоматично створювати категорії та сортувати особисті колекції медіа файлів, відзначати тегами продукцію в онлайн-каталогах, готувати вступні дані для алгоритмів комп'ютерного зору та вирішувати інші завдання в різних сферах – від електронної комерції до допомоги людям з інвалідністю.

Крім вкладу в прогрес в області ШІ, генерація зображень закриває важливі потреби сучасного бізнесу – можливість отримати унікальну картинку під власний опис, а також будь-якої миті створювати необхідну кількість licence-free-ілюстрацій [4]. При цьому створення «мультимодальних» нейронних мереж, які навчаються відразу на декількох видах даних, навіть зараз, в епоху big data та величезних можливостей пошуку, буде дуже затребуваним, оскільки вирішуватиме завдання на принципово іншому рівні. Технологія поки що зовсім нова, перші кроки в цьому напрямку були зроблені лише у рік-два тому.

З появою генеративних змагальних мереж синтез зображень із текстових описів останнім часом стала активною дослідницькою областю. Це гнучкий та інтуїтивно зрозумілий спосіб створення умовного зображення зі значним прогресом останніми роками щодо візуального реалізму, різноманітності та семантичного вирівнювання [5]. Однак ця галузь все ще стоїть перед кількома проблемами, які потребують подальших дослідницьких зусиль, таких як можливість створення зображень високої роздільної здатності з кількома об’єктами, і розробка відповідних і надійних показників оцінки, які корелюють з людськими судженнями. У цій темі є аналіз контексту стана мистецтва змагальних моделей синтезу тексту в зображення, їх розвиток з моменту їх створення

роками тому. Продемонстровано критичний розгляд поточних стратегій для оцінки моделей синтезу тексту в зображення, виділено недоліки та визначанні нові напрямки досліджень, починаючи від розробки кращих наборів даних та показників оцінки для можливих покращень в архітектурному проектуванні та навчанні моделей. Існує багато оглядів доповнює попередніх опитувань щодо генеративних змагальних мереж з акцентом на синтезі тексту в зображення, який на мою думку, допоможе дослідникам у подальшому просуванні цієї галузі.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ ДОСЛІДЖЕННЯ

1.1 Передпроектне обстеження

1.1.1 Опис предметної області

Люди будують знання у зображеннях. Щоразу, коли нам дається ідея чи досвід, наш мозок негайно формулює візуальні уявлення про це. Так само наш мозок постійно перемикається між сенсорними сигналами, такими як звук або текстура, та їх візуальними уявленнями. Наша здатність мислити у візуальних уявленнях не зовсім розширилася до алгоритмів штучного інтелекту. Сьогодні більшість моделей штучного інтелекту сильно спеціалізуються на одній формі представлення даних, як-от зображення, текст або звук. Зрештою, ми почнемо бачити форми ШІ, які можуть ефективно перекладати дані між різними форматами даних, щоб оптимізувати створення знань.

Коли люди чують або читають історію, вони негайно малюють ментальні картини, візуалізуючи вміст у їхній голові. Здатність візуалізувати і розуміти складні відносини між візуальним світом і мовою настільки природна [6], що ми рідко замислюємося про це. Наочні ментальні образи або «бачити розумовим оком» також грає важливу роль у багатьох пізнавальних процесах, таких як пам'ять, просторові навігація та міркування. Натхнений тим, як люди візуалізують сцени, створюючи систему, яка розуміє взаємозв’язок між баченням і мовою і здатна створити зображення, що відображають зміст текстових описів, є важливою віхою на шляху до людського інтелекту. В останні кілька років програми комп'ютерного зору і методи обробки зображень отримали значну користь досягнення, зумовлені проривом глибокого навчання. Одним з них є область синтезу зображень, яка є процесом створення нових зображень і маніпулювання існуючими. Синтез зображень є цікавим і

важливим завданням через багато практичних застосувань, таких як генерація мистецтва, редагування зображень, віртуальна реальність, відеоігри, та комп'ютерне проектування.

Наша здатність генерувати візуальні уявлення з вокальних чи текстових описів є одним із магічних елементів людського пізнання. Якщо нас попросять намалювати зображення баскетбольної гри, ми, ймовірно, почнемо з контуру трьох або чотирьох гравців, які розташовані в центрі полотна. Навіть якщо це не вказано прямо, ми можете додати такі деталі, як ворона, суддя чи гравець у певній позиції для стрілянини. Всі ці деталі збагачують основний текстовий опис, щоб виконати нашу візуальну версію гри у баскетбол. Хіба не було б чудово, якби моделі штучного інтелекту могли робити те саме? Text-to-Image – одна з нових дисциплін глибокого навчання, яка фокусується на створенні зображень із базових текстових уявлень [7]. У той час, як простір TTI знаходиться на дуже ранніх стадіях, ми вже бачимо певний відчутний прогрес з деякими моделями, які довели свою майстерність у дуже специфічних сценаріях. Тим не менш, це дуже специфічні проблеми в моделях, які все ще потребують вирішення.

1.1.2 Проблеми та міркування щодо перетворення

Техніка створення зображення з поданого текстового опису називається синтезом тексту в зображення. Люди можуть візуалізувати та концептуалізувати образи, описані природною мовою. Це, здається, просте завдання для людського мозку, є однією з найскладніших проблем обробки природної мови та комп’ютерного зору [8]. Цифровий світ переповнений різноманітними зображеннями. Але отримати зображення необхідної роздільної здатності, масштабу чи режиму – це далека мрія. Зробити передові медичні зображення або супутникові знімки необхідної якості дуже складно через складність. Синтез тексту в зображення є

благословенням для таких галузей. Деякі інші галузі, такі як дизайн одягу, дизайн інтер’єру тощо, є областями застосування синтезу тексту в зображення. Це вважається складною проблемою порівняно із зворотною проблемою підписання зображень через її мультимодальний характер та складність, пов’язану з вирішенням проблеми. З появою генеративних змагальних мереж (GAN) синтез тексту в зображення досяг нових вершин успіху і привернув велику увагу. У літературі було запропоновано багато моделей синтезу зображень, де GAN відіграють ключову роль.

Існує кілька актуальних проблем, які традиційно блокували еволюцію моделей, але більшість їх можна віднести до однієї з наступних груп [9].

Виклик залежності. Очевидно, що моделі TTI сильно залежать як від текстових, так і від візуальних методів аналізу, які, хоча вони й досягли значного прогресу в останні роки, мають багато роботи, щоб домогтися масового впровадження. З цього погляду можливості моделей TTI, як правило, обмежені специфікою базового аналізу тексту та моделей генерації зображень.

Концептуально-об'єктні відносини. Неймовірно важка проблема, яка має бути вирішена у моделях TTI, – це відносини між концепцією, витягнутою з текстового опису та відповідними візуальними об'єктами. Насправді це може бути безліч об'єктів, відповідних конкретному текстовому опису. З'ясування правильної відповідності між концепціями та об'єктами залишається ключовою проблемою у моделях TTI.

Відношення об'єкт-об'єкт. Будь-яке зображення виражає відносини між об'єктами у візуальному форматі. Щоб відобразити це, модель TTI повинна була не тільки генерувати правильні об'єкти, а й відносини між ними. Створення складніших сцен, що містять кілька об'єктів з семантично значущими зв'язками між цими об'єктами, залишається серйозною проблемою технології генерації тексту в зображення, але в наш час є багато різноманітних статей, які мають на меті дати уявлення про

революційні моделі в області синтезу тексту в зображення.

1.2 Аналіз існуючих аналогів задачі перетворення

Минулий – 2021 рік у машинному навчанні ознаменувався мультимодальністю – активно розвиваються нейромережі, що працюють одночасно із зображеннями, текстами, мовленням, музикою. Править балом, як завжди, OpenAI, але, незважаючи на слово «open» у своїй назві, не поспішає викладати моделі у відкритий доступ. На початку року компанія представила нейромережу DALL-E, що генерує будь-які зображення розміром 256x256 пікселів за текстовим описом [10].

З моменту виходу DALL-E до проблеми активно підключилися китайські дослідники: відкритий код нейромережі CogView дозволяє вирішувати те саме завдання – отримувати зображення з текстів. Але що у світі? Розібрати, зрозуміти, навчити вже, можна сказати, наш інженерний девіз. У проекті брали активну участь команди Sber AI, SberDevices, Самарського університету, AIRI та SberCloud.

В основі архітектури DALL-E – так званий трансформер, він складається з енкодера та декодера. Загальна ідея полягає в тому, щоб обчислити embedding за вхідними даними за допомогою енкодера, а потім з урахуванням відомого виходу правильно декодувати цей embedding.

Основу архітектури трансформера складає механізм Self-attention. Він дозволяє моделі зрозуміти, які фрагменти вхідних даних важливі і наскільки важливим є кожен фрагмент вхідних даних для інших фрагментів. Як і LSTM-моделі, трансформер дозволяє природним чином моделювати зв'язки «довго». Однак, на відміну від LSTM-моделей, він підходить для розпаралелювання і, отже, ефективних реалізацій.

Першим кроком при обчисленні Self-attention є створення трьох векторів для кожного вектора вхідного енкодера (для кожного елемента

вхідної послідовності). Тобто для кожного елемента створюються вектори Query, Key та Value. Ці вектори виходять шляхом перемноження embedding'a та трьох матриць, які ми отримуємо у процесі навчання. Далі використовуються отримані вектори для формування Self-attention представлення кожного embedding'у, що дає можливість оцінити можливі зв'язки в елементах вхідних даних, а також визначити рівень корисності кожного елемента.

Трансформер також характеризує наявність словника. Кожен елемент словника це токен. Залежно від моделі розмір словника може змінюватись. Таким чином, вхідні дані спочатку перетворюються на послідовність токенів, яка далі конвертується в embedding за допомогою енкодера. Для тексту використовується свій токенізатор, для зображення спочатку обчислюються low-level-фічі, а потім у вікні, що ковзає, обчислюються візуальні токени. Застосування механізму Self-attention дозволяє витягти контекст із вхідної послідовності токенів під час навчання. Слід зазначити, що з навчання трансформера потрібні великі обсяги (бажано «чистих») даних, про які ми розповімо нижче.

Отже, Ощад створив першу у світі нейронну мережу ruDALL-E, яка здатна створювати зображення на основі текстового опису російською мовою. Використовувати її можна для створення варіантів дизайну інтер'єру, стічних зображень або векторних ілюстрацій, матеріалів для реклами, копірайтингу, архітектурного та промислового дизайну.

Нейронна мережа одночасно навчається на двох видах даних – картинках і текстах, і дозволяє створювати необмежену кількість нових зображень за заданим описом. Існує два варіанти моделі.

Перше – це ruDALL-E XL, що містить 1,3 мільярда параметрів. Модель RuDALL-E XL можна скористатися безкоштовно.

Створення зображень за допомогою ruDALL-E відбувається в три етапи: спочатку одна нейромережа приймає текст на вхід і генерує задану кількість картинок, потім наступна вибирає, які найбільш вдалі і

максимально відповідають опису, а третя збільшує їх у розмірі без втрати якості. Таким чином, можна отримати необмежену кількість нових зображень, що підходять під зазначені характеристики.

Друге це – ruDALL-E 12B з 12 мільярдами параметрів. Архітектура моделі DALL-E для англійської мови була вперше представлена OpenAI у 2021 році, проте ця модель так і не була повністю викладена у відкритий доступ. На основі публікації OpenAI команди SberDevices і Sber AI за допомогою SberCloud відтворили код і запустили навчання нейромережі на платформі ML Space на базі суперкомп'ютера Крістофарі, отримавши аналогічний результат для російської мови. В результаті вийшла найбільша модель такого роду у світі, що працює з російською мовою: навчання зайняло 23 тисячі GPU-годин на масиві даних із 120 мільйонів пар текст-зображення. Проект з навчання ruDALL-E став найбільшим нейромережевим обчислювальним проектом у СНГ.

1.3 Постановка задачі дослідження

Після проведення аналізу предметної області та існуючих аналогів, необхідно зробити дослідження методів генерації зображень на основі текстових описів та розробити програмний додаток для демонстрації отриманих результатів.

Поточні методи створення стилізованих зображень із текстових описів (тобто нашого базового сценарію) спочатку генеруються зображенням з тексту за допомогою системи GAN, а потім потрібно стилізувати результати за допомогою нейронної передачі стилю. Це займає тривалий час, через використання нейронної передачі стилю, і не буде придатним для використання на низькопродуктивних комп’ютерах. Використання системи лише GAN здається бажаним альтернативним методом для досягнення тієї ж мети набагато меншою кількістю часу.

2 ОГЛЯД ІСНУЮЧИХ НЕЙРОМЕРЕЖЕВИХ МЕТОДІВ ДЛЯ ПЕРЕТВОРЕННЯ ТЕКСТОВОГО ОПИСУ В ЗОБРАЖЕННЯ

Автоматичний синтез реалістичних зображень з тексту став популярним серед глибоко пакувальних та рекурентних архітектур нейронних мереж, щоб допомогти у вивченні розрізняючих уявлень текстових функцій. Дискримінаційна сила та сильні властивості узагальнення уявлень атрибутів, хоч вони й привабливі, але це складний процес і потребує знань, специфічних для предметної галузі [11]. У порівнянні, природна мова пропонує загальний і гнучкий інтерфейс для опису об'єктів в будь-якому просторі візуальних категорій.

2.1 Фундаментальні методи

Генеративний змагальний текст у зображенні Synthesis. Цей механізм синтезу зображень використовує глибокі згорткові та рекурентні кодувальники тексту вивчення функції відповідності із зображеннями, обумовлюючи умови моделі на текстові описи замість міток класів [12]. Ефективний підхід до текстового синтезу зображень з використанням текстового кодувальника на рівні символів та умовного класу GAN.

Оригінальний GAN, складається з двох нейронних мереж: генераторної мережі $G(z)$ із шумом $z \sim pz$ вибірки з попереднього розподілу шуму, і мережу дискримінатора $D(x)$, де $x \sim pdata$ є реальними, а $x \sim pg$ є створені зображення, відповідно. Навчання сформульовано як гра для двох гравців, у якій дискримінатор навчається розрізняти реальні та створені зображення, тоді як генератор навчений фіксувати реальний розподіл даних і створювати зображення, щоб обдурити дискримінатора. Більш формально, навчання можна визначити як мінімаксна гра для двох гравців із функцією значення $V(D, G)$ [13], де дискримінатор $D(x)$

навчається максимізувати логарифмічну ймовірність, яку він призначає правильному класу, тоді як генератор *G(z)* навчений мінімізувати ймовірність того, що дискримінатор *log(1−D(G(z)))* класифікує як підробку. *D(G(z))*, функція втрат позначається як *Ladv* у наших фігурах. Зразок формування тексту наведено на рисунку 2.1.

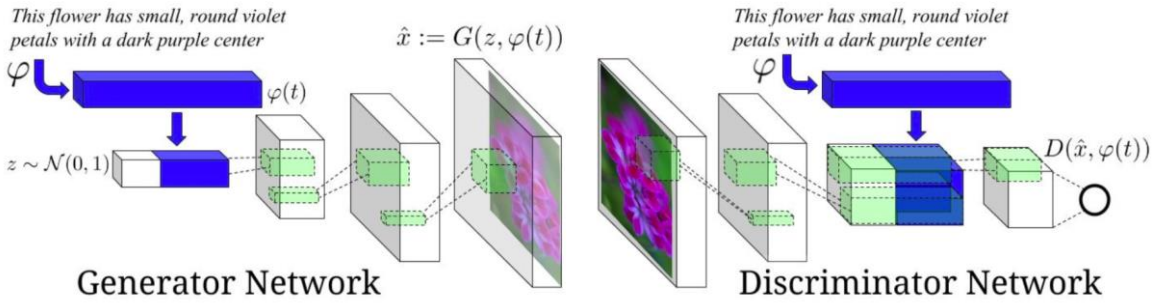

Рисунок 2.1 – Зразок формування тексту

Текстова умовно-згорткова архітектура GAN, де текстове кодування *ϕ(t)* використовується як генератором, так і дискримінатором. Він проектується з меншими розмірами та глибиною, поєднаними з картами характеристик зображення для подальших етапів згорткової обробки. Текстова умовно-згорткова архітектура GAN наведена на рисунку 2.2.

Рисунок 2.2 – Текстова умовно-згорткова архітектура GAN

Доповнення до реальних/фальшивих входів у дискримінатор під час

навчання він також забезпечується входом третього типу, що складається з реальних зображень з невідповідним текстом, що допомагає дискримінатору оцінювати його як фальшивий. Перенесення стилю з зображень верхнього рядка (реальних) до контенту з тексту запиту, де *G* виступає як детермінований декодер. Три нижні рядки це підписи, складені нами.

Умовні GAN. Умовні GAN. Хоча створення нових, реалістичних зразків є цікавим, отримання контролю над процесом генерації зображення має високу практичну цінність. Mirza та ін. запропонував [14] умовний GAN (cGAN) шляхом включення змінної кондиціонування у (наприклад, мітки класів) як у генераторі, так і в дискримінаторі, щоб вказати, яку цифру MNIST створити. У їхніх дослідах $z \sim pz$ і у є входами до багатошарової мережі персептронів (MLP) з одним прихованим шаром, утворюючи таким чином спільну приховану представлення для генератора. Аналогічно, для дискримінатора MLP поєднує зображення та мітки.

Кодування тексту. Створення вбудовування з текстових уявлень що корисно для мережі з точки зору змінної умовності, не є тривіальним. Reed та ін. отримали кодування тексту текстового опису за допомогою попередньо навченої згорткової рекурентної нейронної мережі (Char-CNN-RNN) [15]. Char-CNN-RNN попередньо навчений вивчати функцію відповідності між текстом і зображення на основі міток класу. Це призводить до візуально дискримінаційного кодування тексту. Під час навчання додаткового вбудовування тексту було створено шляхом простої інтерполяції між вбудовуваннями двох навчальних підписів. Автори також показали, що традиційні текстові уявлення такі як Word2Vec і Bag-of-Words були менш ефективні. TAC-GAN використовував вектори пропуску думки. Замість використання фіксованого вбудовування тексту, отриманого за допомогою попередньо навченого текстового кодеру, автори StackGAN запропонували посилення кондиціонування (CA) [16], де випадковим чином вибірка латентної змінної з гауссового розподілу, де

середнє і коваріаційна матриця є функціями від вбудовування тексту. Розбіжність Кульбака-Лейблера між стандартним розподілом Гаусса та умовний гауссовий розподіл використовується як термін регуляризації під час навчання. Ця техніка дає більше тренувальних пар і сприяє плавності на колекторі кондиціонування. Багато з наступних методів T2I взяли на озброєння цю техніку.

Датасети. Набори даних є ядром кожної проблеми машинного навчання. Широко поширеними наборами даних у дослідженнях T2I є Oxford120 Flowers, CUB-200 Birds та COCO [17]. Oxford-102 Flowers, і CUB-200 Birds є відносно невеликими наборами даних, що містять близько 10 тисяч зображень. Кожне зображення зображує один об’єкт, і для кожного зображення є десять пов’язаних підписів. COCO з іншого боку складається з приблизно 123 тисяч зображень із п’ятьма підписами на зображення. На відміну від Oxford-102 Flowers і CUB-200 Birds, зображення в наборі даних COCO зазвичай містять кілька, часто взаємодіючих об’єктів у складних налаштуваннях. Існуючі датасети наведені на рисунку 2.3.

Рисунок 2.3 – Існуючі датасети

2.2 Прямі методи T2I

Перші підходи T2I. T2I використовує комбіновану архітектуру ProGAN для синтезу зображень особи і StackGAN для кодування тексту з умовним доповненням [18]. Текстовий опис кодується в зведений вектор з використанням мережі LSTM. Зведений вектор, тобто. Вкладення пропускається через блок кондиціонування доповнення (один лінійний шар) щоб отримати текстову частину прихованого вектора (використовує VAE-подібний метод повторної параметризації) для GAN як вхідні дані. Друга частина прихованого вектора випадковий гауссів шум. Вироблений таким чином прихований вектор надходить в генераторну частину GAN, в той час як вкладення подається в останній шар дискримінатора для зіставлення умовного розподілу. Навчання GAN просувається шар за шаром при збільшенні просторових дозволів. Новий шар запроваджено з використанням техніки поступової появи, щоб уникнути руйнування попереднього навчання. Приклад перетворення з T2I методом наведено на рисунку 2.4.

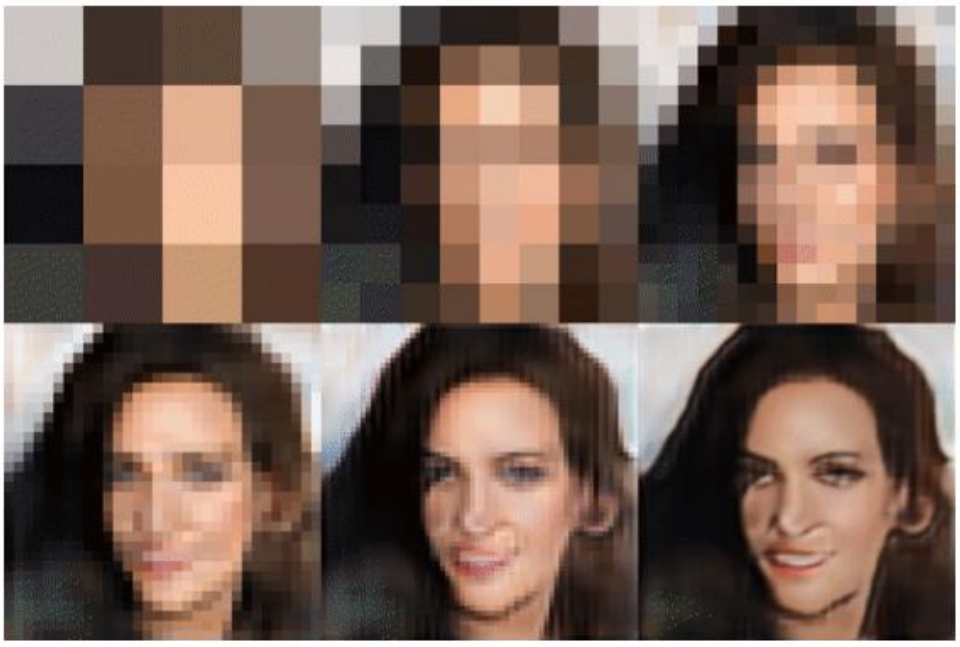

Рисунок 2.4 – Приклад перетворення з T2I методом

Архітектура проекту T2I комбінує дві архітектури stackGAN для кодування тексту з умовним збільшенням і ProGAN для синтезу зображень осіб. Механізм генерації зображення особи з текстових підписів кожного з них. Принцип роботи архітектури T2I наведений на рисунку 2.5.

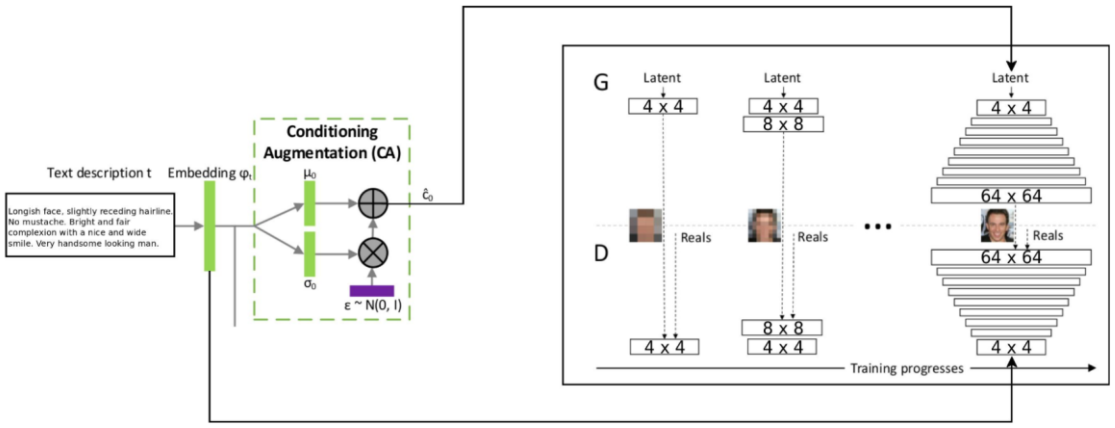

Рисунок 2.5 – Принцип роботи архітектури T2I

Стековані архітектури. GAN-INT-CLS зміг генерувати зображення з низькою роздільною здатністю 64 × 64 пікселів, тоді як TAC-GAN генерував зображення розміром 128 × 128 пікселів. Щоб дозволити моделям T2I синтезувати зображення з вищою роздільною здатністю [19], у багатьох наступних роботах пропонувалося використовувати декілька генераторів із стеком. У StackGAN перший етап генерує грубе зображення розміром 64×64 пікселів із випадковим вектором шуму та вектором текстового кондиціонування. Це початкове зображення та вбудовування тексту, потім вводяться до другого генератора, який виводить зображення розміром 256 × 256 пікселів. На обох етапах дискримінатор навчається розрізняти відповідні та невідповідні пари зображення-текст. StackGAN++ додатково покращив архітектуру за допомогою наскрізної структури, в якій три генератори та дискримінатори спільно навчаються, щоб одночасно апроксимувати багато масштабні, умовні та безумовні розподіли зображень. Автори запропонували взяти зразки вбудовування

тексту з гауссового розподілу для гладкості кондиціонування, замість використання фіксованого вбудовування тексту. Щоб спонукати мережу створювати зображення в кожному масштабі для спільної основної структури та кольорів, був запропонований додатковий термін регуляризації колірної узгодженості, який спрямований на мінімізацію відмінностей між середнім значенням і коваріацією пікселів між різними масштабами. Подібно до ідеї навчання умовних і безумовних розподілів одночасно, FusedGAN складається з двох генераторів (один для безумовного, а другий для умовного синтезу зображень), які частково поділяють спільний латентний простір, щоб дозволити як умовне, так і безумовне генерування з той самий генератор. Щоб подолати потребу в кількох мережах генераторів, HDGAN використовував ієрархічно вкладені дискримінатори на багато масштабних проміжних рівнях для створення зображень 512×512. Іншими словами, змагальна гра ведеться на глибині генератора з окремими дискримінаторами на кожному рівні роздільної здатності. На додаток до втрат усвідомленої пари збігу, дискримінатори також навчені відрізняти реальні плями зображення від створених. Ця мета діє як регуляризатор для прихованих шарів генератора, оскільки виходи на проміжних рівнях можуть використовувати сигнал від дискримінаторів з більш високою роздільною здатністю для отримання більш узгоджених вихідних даних між різними масштабами. Архітектури GAN-INT-CLS, StackGAN та StackGAN++ наведені на рисунках 2.6 та 2.7.

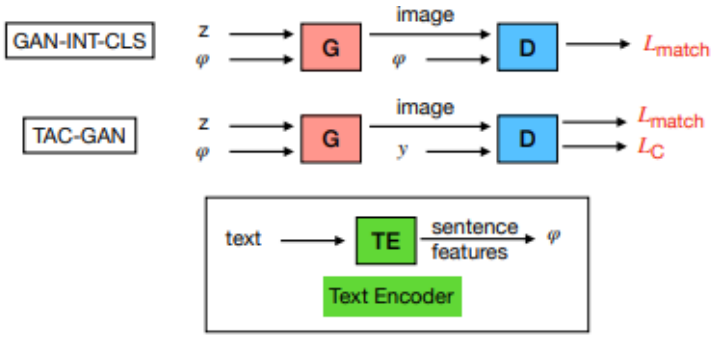

Рисунок 2.6 – Архітектура GAN-INT-CLS

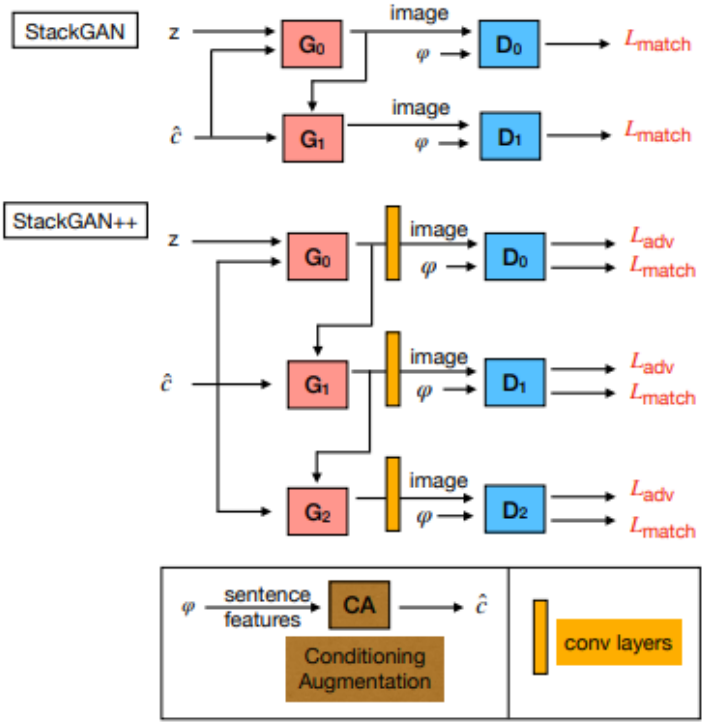

Рисунок 2.7 – Архітектура StackGAN та StackGAN++

Механізми уваги. Техніки уваги дозволяють мережі зосередитися на конкретних аспектах вхідних даних, зважуючи важливі частини більше, ніж неважливі. Увага – це дуже потужна техніка, яка мала великий вплив на покращення мовних і зорових додатків. AttnGAN спирається на StackGAN++ і включає увагу в багатоетапний конвеєр уточнення. Механізм уваги дозволяє мережі синтезувати дрібні деталі на основі відповідних слів на додаток до глобального вектора речення [20]. Під час створення мережі рекомендується зосередитися на найбільш релевантних словах для кожного субрегіону зображення. Це досягається завдяки втраті моделі глибокої уваги мультимодальної схожості (DAMSM) під час навчання, яка обчислює подібність між згенерованим зображенням і введеним текстом, використовуючи інформацію на рівні речення та слова. Розшрена увага на основі сітки з додатковим механізмом між областями об'єктної сітки та словосполученнями, де області сітки об'єктів визначаються допоміжними обмежуючими рамками. Функції фраз

виділяються на додаток до ознак речень і слів шляхом застосування тегів частини мови. Автори SEGAN запропонували модуль змагання уваги, щоб зосередитися лише на ключових словах замість визначення ваги уваги для кожного слова в реченні (як це зроблено в AttnGAN). Вони досягли цього, запровадивши термін регуляризації уваги, який зберігає вагу уваги лише для візуально важливих слів. ControlGAN може виконувати обидва: генерацію T2I та маніпулювання такими візуальними атрибутами, як категорія, текстура та колір, змінюючи опис, не впливаючи на інший вміст (наприклад, фон і пози). Автори запропонували на рівні слова просторовий і канальний генератор уваги, який дозволяє генератору синтезувати області зображення, що відповідають найбільш релевантним словам. У порівнянні з просторовою увагою в, яка в основному зосереджена на інформації про колір, увага по каналу співвідносить семантично значущі частини з відповідними словами (наприклад, «голова» та «крила» для птахів CUB-200). А Дискримінатор на рівні слів забезпечує генератор тонкими навчальними сигналами та роз'єднує різні візуальні атрибути, використовуючи кореляцію між словами та субрегіонами зображення. Архітектура AttnGAN наведена на рисунку 2.8.

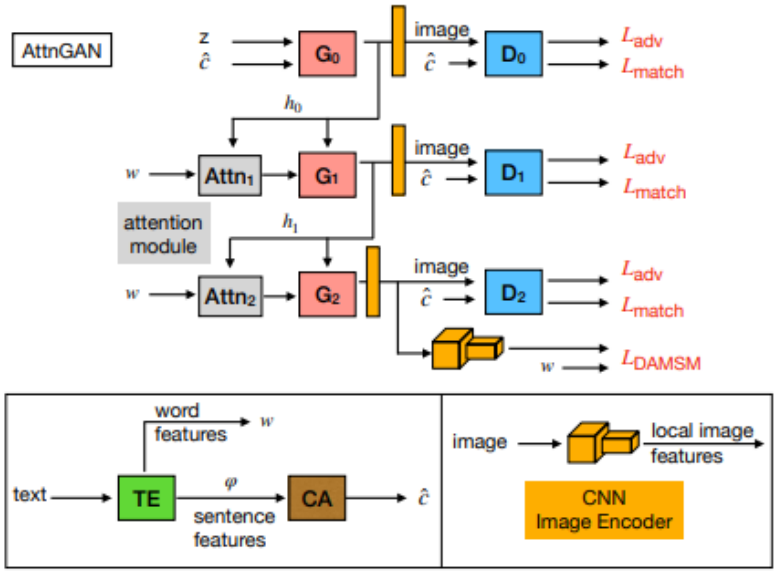

Рисунок 2.8 – Архітектура AttnGAN

Сіамські архітектури. Сіамські мережі, вперше запропоновані для вирішення проблем сигнатури та перевірки обличчя, зазвичай складаються з двох гілок із спільними параметрами моделі, що працюють на парі входів [21]. Кожна гілка працює на різних вхідних даних, і мета полягає в досягненні відображення, де входи зі схожими шаблонами розміщуються ближче один до одного, ніж різні. SD-GAN – це така сіамська мережева архітектура, що складається з двох гілок. У той час як окремі гілки мережі обробляють різні текстові введення для створення файлу зображення, параметри моделі є спільними. Контрастівна втрата, використовується для мінімізації/ максимізації відстані між ознаками, обчисленими в кожній гілці, щоб навчитися семантично значущого представлення, залежно від того, чи є два підписи з одного основного зображення правди (внутрішньокласової пари) чи ні (міжкласна пара). Цей підхід виділяє семантичні спільні елементи з тексту, але може ігнорувати дрібне семантичне різноманіття. Щоб зберегти різноманітність створених зображень, автори додатково запропонували Semantic-Conditioned Batch Normalization, варіант умовної пакетної нормалізації, щоб адаптувати карти візуальних ознак залежно від лінгвістичних сигналів. SEGAN навчає сіамську архітектуру використовувати зображення наземної істини для семантичного вирівнювання. Вони роблять це, мінімізуючи відстань між згенерованим зображенням і відповідним наземним істинним зображенням, максимізуючи відстань до іншого реального зображення, пов’язаного з іншим підписом. Щоб ефективно збалансувати легкі та жорсткі зразки, автори запропонували ковзну втрату, натхненну фокальними втратами, щоб адаптувати відносну важливість пар легких і жорстких зразків. Замість випадкової вибірки невідповідного зразка негативного зображення, у Text-SeGAN введено кілька стратегій, заснованих на навчанні за навчальною програмою, для відбору негативних зразків із поступово зростаючими семантичними труднощами. Замість використання класифікації як допоміжного завдання, автори

сформулювали завдання регресії для оцінки семантичної коректності на основі семантичної відстані до закодованого довідкового тексту. Архітектура SD-GAN наведена на рисунку 2.9.

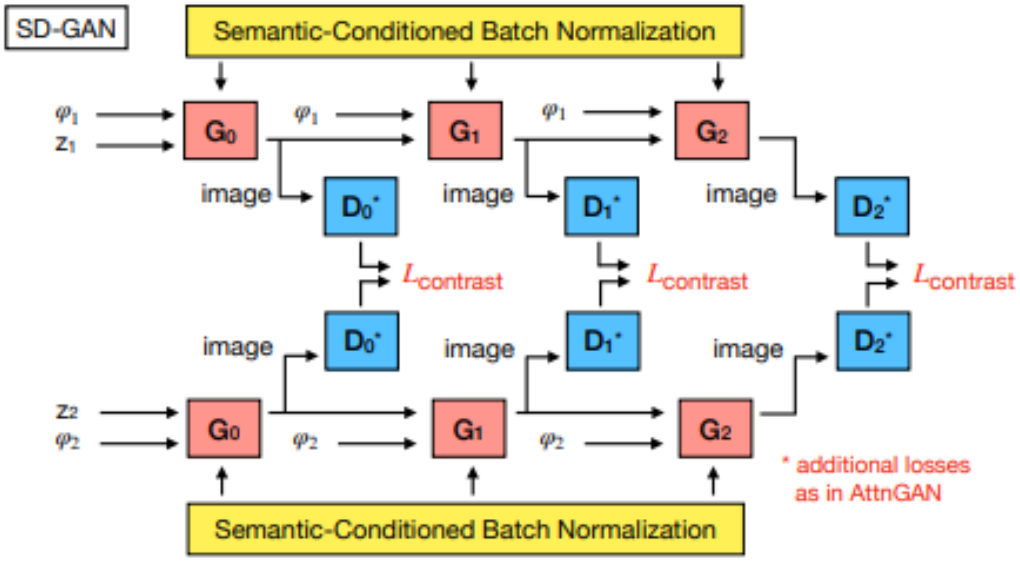

Рисунок 2.9 – Архітектура SD-GAN

Послідовність циклу. Ми групуємо моделі T2I, які беруть згенероване зображення та передають його через підпис зображення або мережу кодувальників зображень, створюючи таким чином цикл до вхідного опису або прихованого коду, у міру наближення узгодженості циклу. PPGN заснований на зворотному зв’язку від умовної мережі, яка може бути або класифікатором, або мережею підписів зображень для умовного синтезу зображень. Основна ідея полягає в тому, щоб ітеративно знайти прихований код, який веде генератор до створення зображення, яке максимізує активацію певної функції в мережі зворотного зв’язку (наприклад, оцінка класифікації або прихований вектор RNN) [22]. У цій структурі попередньо підготовлений генератор можна переназначити, підключивши іншу мережу зворотного зв’язку. Натхненний CycleGAN, циклічне генерування зображень за допомогою архітектур переопису вивчає семантично узгоджене представлення між текстом і зображенням,

додаючи мережу підписів, і навчає мережу створювати семантично подібний підпис із синтезованого зображення. У MirrorGAN вбудовування речень і слів використовуються для керування каскадною архітектурою генератора через глобальне речення та увагу до локальних слів. Далі, мережа субтитрів зображень на основі кодера-декодера використовується для створення підпису на основі згенерованого зображення. На додаток до втрат протилежного збігу зображення та зображення-текст, втрата реконструкції тексту на основі перехресної ентропії використовується для вирівнювання семантики між вхідним підписом і повторним описом. Натхненні змагальними методами висновку запропонували роз'єднати стиль (знятий за допомогою вектора шуму) та вміст (описаний за допомогою вбудовування тексту) у прихованому просторі без нагляду. У їхньому методі додатковий кодер бере реальні зображення та виводить дві приховані змінні (стиль і вміст), які згодом використовуються для створення зображення. Термін втрати узгодженості циклу обмежує кодер і декодер бути узгодженими один з одним. На додаток до суперницької втрати іміджу, вони також використовують дискримінатор, щоб розрізняти спільні пари зображень і прихованих кодів. Архітектура MirrorGAN наведена на рисунку 2.10.

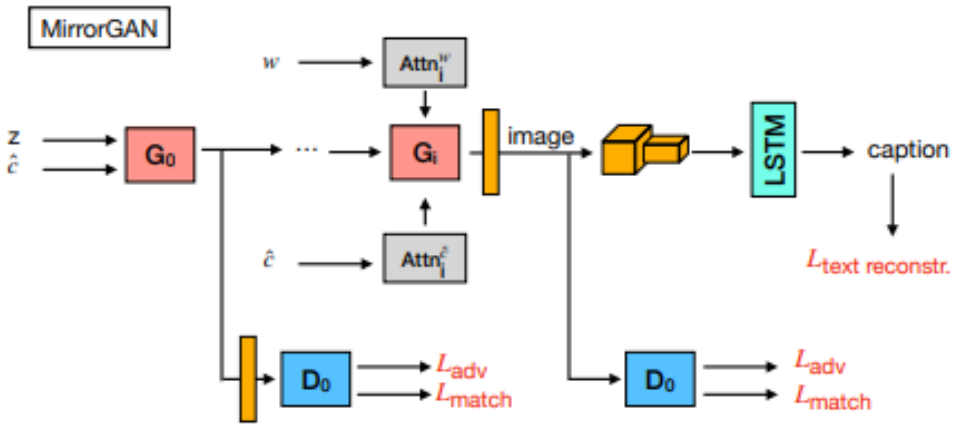

Рисунок 2.10 – Архітектура MirrorGAN

MirrorGAN – це глобально локальний уважний та семантико-зберігаючий фреймворк для перетворення тексту на зображення. MirrorGAN відповідає за навчання генерації тексту в зображення шляхом повторного опису і складається з трьох модулів: модуль вбудовування семантичного тексту (STEM), глобально-локальний уважний модуль для створення каскадних зображень (GLAM), а також модуль регенерації та вирівнювання семантичного тексту (STREAM) [23], На рисунку 2.11 показані окремі компоненти. Компоненти фреймворку MirrorGan наведені на рисунку 2.11.

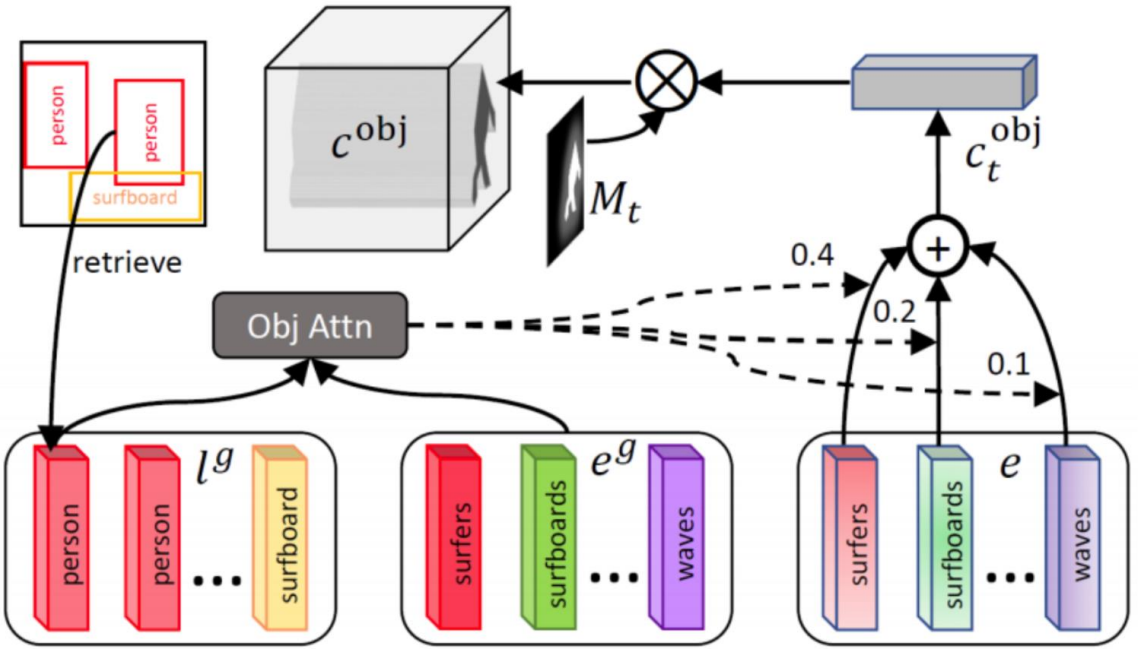

Рисунок 2.11 – Компоненти фреймворку MirrorGan

STEM генерує вкладення на рівні слів та речень, використовуючи рекурентну нейронну мережу (RNN) для вбудовування даного текстового опису в локальні функції на рівні слів та глобальні функції на рівні речень.

GLAM має багатоетапний каскадний генератор, що послідовно поєднує три мережі генерування зображень для генерації цільових зображень від грубих до дрібних масштабів, використовуючи як локальну

увагу до слів, так і глобальну увагу до пропозицій для поступового підвищення різноманітності та семантичної узгодженості згенерованих зображень. STREAM прагне відновити текстовий опис із згенерованого зображення, яке семантично вирівнюється із заданим текстовим описом.

Модель уваги на рівні слів використовується для створення уважної функції контексту слів. Вкладення слів і візуальна особливість приймаються як вхідні дані на кожному етапі. Вкладення слова спочатку перетворюється на базовий загальний семантичний простір візуальних ознак за допомогою рівня сприйняття і множиться на візуальну ознаку для отримання оцінки уваги. Зрештою, функція уважного контексту слова виходить шляхом обчислення внутрішнього твору між показником уваги та рівнем сприйняття разом із вбудовуванням слова.

MirrorGAN включає в себе відновлення семантичного тексту і модуль вирівнювання відновлення текстового опису зі згенерованого зображення, яке семантично вирівнюється із заданим текстовим описом. В архітектурі використовується структура заголовка зображення на основі кодера-декодера. MirrorGAN працює краще, ніж AttnGAN при будь-яких налаштуваннях з великим відривом, демонструючи перевагу запропонованої структури перетворення тексту на зображення і тексту та глобального локального спільного уважного модуля. Принцип роботи STEM, GLAM, STREAM наведений на рисунку 2.12.

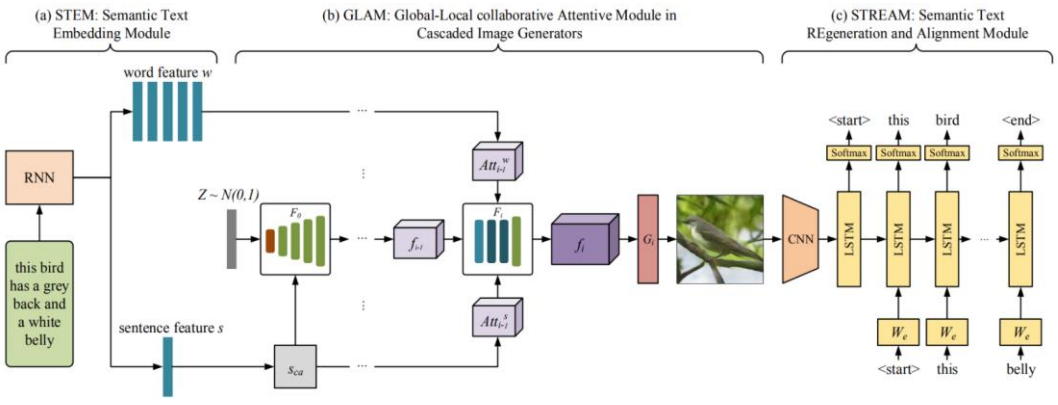

Рисунок 2.12 – Принцип роботи STEM, GLAM, STREAM

Мережі пам’яті. DM-GAN – це архітектура, заснована на мережах динамічної пам’яті. DM-GAN складається з початкового етапу генерації зображення для синтезу грубого зображення розміром 64 × 64 пікселя з урахуванням вбудовування речення. Шлюз запису пам’яті приймає початкові характеристики зображення та слова як вхідні дані, обчислює важливість кожного слова і, нарешті, записує слоти пам’яті, поєднуючи функції слова та зображення [24]. Потім виконується крок адресації ключа та зчитування значень, на якому відповідні слоти пам’яті витягуються шляхом обчислення ймовірності подібності між слотами пам’яті та характеристиками зображення. Після цього представлення вихідної пам’яті обчислюється шляхом зваженого підсумовування пам’яті значень відповідно до подібності ймовірність. Нарешті, стробована відповідь динамічно контролює потік інформації вихідного представлення для оновлення характеристик зображення. Подібно до обговорених раніше моделей T2I, DM-GAN використовує безумовне змагальне зображення та умовні втрати узгодження зображення-текст. Крім того, використовуються втрати DAMSM і втрати CA.

Адаптація безумовних моделей. Спираючись на прогрес у безумовній генерації зображень, у багатьох роботах було запропоновано адаптувати архітектуру цих безумовних моделей для умовної генерації T2I. Автори textStyleGAN розширили StyleGAN [25], який може генерувати зображення з більш високою роздільною здатністю, ніж інші моделі T2I, і дозволяє здійснювати семантичні маніпуляції. Автори запропонували обчислити вбудовування тексту та слів, використовуючи попередньо навчену мережу зіставлення зображення-текст, подібну до тієї, що використовується в AttnGAN, і об’єднати вбудовування речення з вектором шуму перед виконанням лінійного відображення для отримання проміжного латентний простір. Крім того, вони використовують наведення уваги, використовуючи функції слів і зображень у генераторі. На додаток до безумовних та умовних втрат у дискримінаторі, втрати крос-модального

узгодження проекції (CMPM) та міжмодальної класифікації проекції (CMPC) використовуються для узгодження вхідних підписів із згенерованими зображеннями. Маніпуляцію зображенням можна виконати, спочатку знайшовши напрямки в проміжному прихованому просторі, що відповідають семантичним атрибутам, таким як «усмішка» та «вік» для зображень обличчя. Оскільки проміжний латентний простір у StyleGAN не повинен підтримувати вибірку, було емпірично показано, щоб розкрутити початковий прихований код таким чином, що фактори варіації стають більш лінійними і, як наслідок, підтримують маніпулювання семантичними зображеннями. Архітектура StyleGAN наведена на рисунку 2.13.

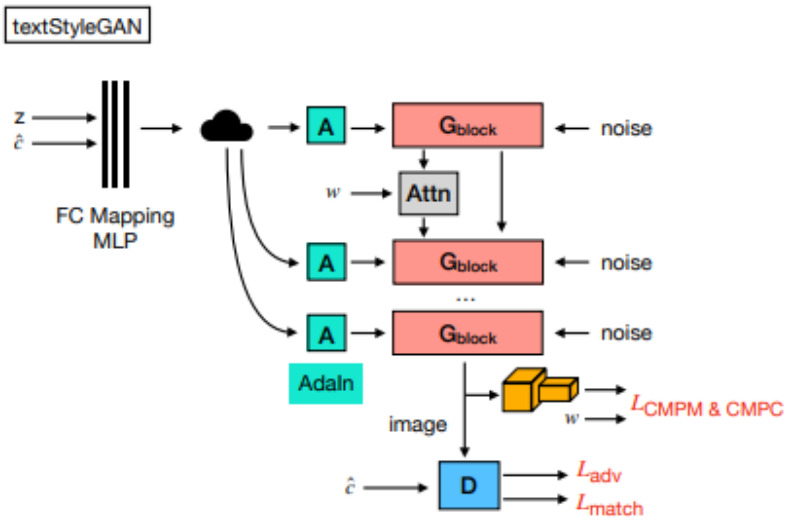

Рисунок 2.13 – Архітектура StyleGAN

Bridge-GAN використовує схему прогресивного зростання генератора та дискримінатора під час навчання. За мотивами, проміжна мережа використовується для відображення вбудовування тексту та шуму в перехідний простір відображення, і пропонуються дві додаткові втрати на основі взаємної інформації [26]. Перша втрата обчислює взаємну інформацію між проміжним прихованим простором і вбудованим текстом, щоб гарантувати наявність текстової інформації в перехідному просторі.

Друга втрата обчислює взаємну інформацію між згенерованим зображенням і введеним текстом, щоб покращити узгодженість між зображенням і введеним текстом. У роботі автори адаптували BigGAN, архітектуру, яка раніше представляла нову сучасну технологію в ImageNet, яка обумовлена мітками класів, для синтезу T2I. Крім того, вони запропонували новий метод інтерполяції речень (SI) для створення інтерпольованих вкладень речень з використанням усіх доступних підписів, що відповідають певному зображенню. У порівнянні з CA, який вводить випадковість і оптимізує дивергенцію KL для забезпечення гауссового розподілу, SI є детермінованою функцією. Аналогічно, TVBi-GAN використовував архітектуру BigGAN, розширюючи визначення прихованого простору в ALI, щоб проектувати в нього особливості речення. Крім того, автори запропонували механізм воріт, натхненний, щоб обчислити важливість між характеристиками слова та семантичними ознаками перед тим, як привернути увагу. Крім того, пропонується посилена семантика пакетна нормалізація, шляхом введення випадкового шуму для стабілізації операції масштабування та зсуву на основі лінгвістичних сигналів. Автори навчають інверсійну мережу для злиття попередньо підготовлених експертних мереж BERT і BigGAN, перекладу між їхніми уявленнями та повторного використання їх для синтезу тексту в зображення. Це дуже перспективний напрямок досліджень для повторного використання експертних мереж, які дорого навчати для інших завдань.

2.3 Методи T2I з додатковим наглядом

Кілька субтитрів. Оскільки звичайні набори даних часто містять більше одного підпису на зображення, використання кількох підписів може надати додаткову інформацію для кращого опису всієї сцени [27]. C4Synth використовує декілька підписів, використовуючи узгодженість циклу перехресних підписів, яка гарантує, що створене зображення

узгоджується з набором семантично подібних речень. Він працює послідовно, перебираючи всі підписи, і покращує якість зображення шляхом виділення концепцій із кількох підписів. RiFeGAN розглядає доступні зображення та підписи як базу знань і використовує механізм відповідності підписів для отримання сумісних елементів. Вони збагачують вхідний опис, витягуючи функції з кількох підписів, щоб направляти генератор зображень уваги. RiFeGAN не потребує мережі підписів зображень і виконується один раз замість кількох разів.

Діалог. Мотивований тим, що одне речення може бути недостатньо інформативним, щоб описати сцену, яка містить кілька взаємодіючих об’єктів, Вчені запропонували ChatPainter для використання діалогових даних. Автори використовують набір даних Visual Dialog, який складається з 10 поворотів розмови запитання-відповідь на діалог, і поєднують його з підписами COCO [28]. Автори експериментували з рекурентним і нерекурентним кодером і показали, що рекурентний кодер працює краще. Інші вчені запропонували VQA-GAN обумовити генератор зображень для локально пов’язаних текстів за допомогою пар запитань-відповідей (QA) з VQA 2.0, набору даних, побудованого на COCO для завдань візуальних відповідей на запитання (VQA). Їх метод заснований на AttnGAN-OP і складається з трьох ключових компонентів: i) кодер QA, який приймає QA пари як вхідні дані для створення глобальних і локальних уявлень, ii) обумовлений QA GAN, який бере уявлення від кодера QA для створення зображення в двоетапному процесі, iii) зовнішні втрати VQA за допомогою моделі VQA що сприяє кореляції між парами QA та створеним зображенням. Типова модель VQA приймає зображення та запитання як вхідні дані та навчається для класифікації, тобто для мінімізації негативної втрати логарифмічної правдоподібності, щоб максимізувати ймовірність правильної відповіді. Отже, точність VQA можна використовувати як метрику для оцінки узгодженості між вхідними парами QA та згенерованими зображеннями. На додаток до пар QA з VQA 2.0, їхня

модель також потребує нагляду в форма макета. Автори запропонували використовувати дані VQA без зміни архітектури. Просто об’єднавши пари QA та використавши їх як додаткові навчальні зразки та втрату зовнішнього VQA, можна підвищити продуктивність як якості зображення, так і показників вирівнювання зображення-текст. Це проста, але ефективна методика, яку можна застосувати до будь-якої моделі T2I.

Макет. Зростає інтерес до завдання генерації макета до зображення, де кожен об’єкт визначається обмежувальною рамкою та міткою класу. Це надає більше структури генератору, сприяє кращому локалізації об’єктів у зображенні та має перевагу, що дозволяє контролювати користувачеві генерацію шляхом зміни макета, а створені зображення автоматично коментуються. Природно, дослідники також намагалися поєднати інформацію про макет з текстом для кращого T2I. GAWWN [29] визначає як текстові описи, так і розташування об’єктів, щоб продемонструвати ефективність цього підходу на наборі даних CUB-200 Birds. Наступна робота розширює PixelCNN для створення зображень із підписів із контрольованими розташуваннями об’єктів, використовуючи ключові точки та маски. Для більш ефективного висновку використовується розпаралелізований PixelCNN. Розташування та зовнішній вигляд об'єктів явно моделюється шляхом додавання шляху об'єкта як до генератора, так і до дискримінатора. У той час як шлях об’єкта зосереджується на створенні окремих об’єктів у значущих місцях, глобальний шлях створює фон, який відповідає загальному опису та макету зображення. OP-GAN розширює це, додаючи додаткові шляхи об'єкта на вищих рівнях генератора та дискримінатора, а також використовує додаткову обмежувальну рамку узгодження втрат, використовуючи пари зображень узгоджених і невідповідних обмежувальних рамок. OCGAN вирішує проблему об'єднаних об'єктів і помилкових режимів, пропонуючи модуль подібності сцен-графів (SGSM), подібний до DAMSM в AttnGAN.

Семантичні маски. Інша лінія досліджень використовує маски для

вивчення форм об’єктів, тим самим забезпечуючи ще кращий сигнал мережі. Вчені отримують семантичні маски в двоетапний процес: перший крок генерує макет із вхідного опису, який потім використовується для прогнозування форм об’єкта. Він має одноступінчастий генератор зображень і обумовлює лише згенеровану форму та глобальне речення інформації. Obj-GAN спирається і складається з об'єктно-керованого генератора уважності та об'єктного дискримінатору. Генератор використовує GloVe [30] вбудовування міток класу об’єктів, щоб запитувати вбудовування GloVe відповідних слів у реченні. Об’єктний дискримінатор заснований на Fast R-CNN для надання сигналу про те, чи є синтезовані об’єкти реалістичними та відповідають макету та текстовому опису. LeicaGAN має фазу навчання кількох попередніх етапів, на якій кодер текстових зображень вивчає семантичні, текстурні та колірні пріоритети, тоді як кодер маски тексту вивчає форму та макет. Ці додаткові пріоритети об’єднуються та використовуються для використання як локальних, так і глобальних функцій для поступового створення іміджу. Щоб зменшити розрив домену під час проеціювання вхідного тексту в основний загальний простір, автори прийняли під час навчання класифікатор модальності, навчений протиборцями. AGAN-CL складається з мережі, яка навчена створювати маски, надаючи таким чином дрібнозернисту інформацію, таку як кількість об’єктів, розташування, розмір і форма. Автори використали багатомасштабну втрату між реальною та створеною масками, а також додаткове сприйняття втрата глобальної узгодженості. На наступному кроці маска зображення подається в якості входу в циклічний автокодер, для створення фотореалістичного зображення. У роботі вчений запропонував наскрізну структуру з просторовими обмеженнями, використовуючи семантичний макет, щоб керувати синтез зображень. Багатомасштабні семантичні макети об’єднані з семантикою тексту та прихованими візуальними функціями, щоб створити зображення від грубого до точного. На кожному

етапі генератор створює зображення і додатково макет, який буде використовуватися відповідним дискримінатором. Дискримінатор з узгодженням із розширено, щоб також розрізняти пари макета та тексту, що збігаються, а також розрізняти реальні макети від згенерованих. Вчений запропонував підхід із слабким контролем, використовуючи розріджені семантичні маски екземплярів. На відміну від масок на основі щільних пікселів, маски розріджених екземплярів дозволяють легко виконувати операції редагування, такі як додавання або видалення об’єктів, оскільки користувач не стикається з проблемою «заповнення цілого». Їх метод особливо хороший для контролю дрібнозернистих деталей окремих об’єктів, що реалізується двоетапним процесом генерації, який розкладає фон з переднього плану.

Графи сцен. Відносини між кількома об’єктами часто можуть бути більш чітко представлені структурованим текстом, тобто графіком сцени замість підпису. Для COCO, де анотації графіка сцени не надаються, графік сцени можна побудувати з розташування об’єктів, використовуючи шість геометричних відношень: «ліворуч», «праворуч», «зверху», «внизу», «всередині» та « навколишні». Однак існують також інші набори даних з більш дрібними анотаціями графа сцени, що робить цей підхід дуже перспективним (наприклад, Visual Genome [31] забезпечує в середньому 21 попарний зв’язок на зображення). Вчений використовував нейронну мережу графа для обробки вхідних графів сцени і обчислив макет сцени, передбачивши обмежувальні рамки та маски сегментації для кожного об’єкта. Окремі блоки та маски об’єктів об’єднуються для формування макета сцени, а потім використовуються для створення зображення каскадною мережею уточнення. Під час навчання використовуються обмежувальні рамки та додаткові маски, але передбачені під час тестування. Розширенням, яке використовує маски сегментації. Він відокремлює вбудовування макета від вбудовування зовнішнього вигляду, що сприяє кращому контролю з боку користувачі та створені зображення,

які краще відповідають графіку вхідної сцени. Атрибути зовнішнього вигляду можна вибрати із попередньо визначеного набору або скопіювати з іншого зображення. Графік сцени використовується для прогнозування початкових обмежувальних рамок для об’єктів. Використовуючи початкові рамки, для кожного окремого відношення суб’єкт-присудок-об’єкт прогнозуються одиниці відношень, що складаються з двох обмежувальних прямокутників. Оскільки кожна сутність може брати участь у кількох відносинах, усі одиниці відношень уніфікуються та перетворюються у макет візуального відношення за допомогою згорткового LSTM. Візуально-відносний макет відображає структуру (об’єкти і відносини) на графі сцени, і кожна сутність відповідає одній уточненій обмежувальній рамці. Нарешті, макет візуального відношення використовується в умовній архітектурі GAN з стекуванням для відтворення кінцевого зображення. PasteGAN використовує графіки сцени та обрізання об’єктів для керування процесом створення зображення. У той час як графік сцени кодує просторове розташування та взаємодії, зовнішній вигляд кожного об’єкта забезпечується заданими об’єктами. Об’єктні кадрування та зв’язки об’єднуються разом, а потім передаються в декодер зображення для створення вихідного зображення. Інтерактивна структура розширюється за допомогою а рекурентна архітектура для створення послідовних зображень із поступово зростаючого графіка сцени. Модель оновлює зображення, згенероване з графіка сцени, змінюючи графік сцени, максимально зберігаючи раніше згенерований вміст. Збереження попереднього зображення заохочується заміною шуму, переданого в каскадний генератор зображень, попереднім зображенням і додатковою втратою сприйняття між зображеннями на проміжних кроках.

Мишачі сліди. TRECS [32] використовує сліди миші, зібрані людьми-анотаторами в наборі даних Localized Narratives, який поєднує зображення з докладними описами природної мови та слідами миші. Сліди миші забезпечують рідкісне, дрібнозернисте візуальне обґрунтування для

описів. З огляду на численні описи та відповідні сліди миші, TRECS отримує семантичні маски, з яких генеруються зображення.

2.4 Інші нейромережеві методи для задачі перетворення

Multi-Scale Gradient GAN для стабільного синтезу зображень. Мультимасштабна градієнтна генеративна змагальна мережа відповідає за обробку нестабільності в градієнтах, що переходять від дискримінатора до генератора, які стають неінформативними через дисбаланс навчання під час навчання [33]. Він використовує ефективну техніку, яка дозволяє передавати градієнти від дискримінатора до генератора у кількох масштабах, допомагаючи генерувати синхронізовані багатомасштабні зображення. Дискримінатор дивиться не тільки на кінцевий вихід (найвища роздільна здатність) генератора, але також на виходи проміжних шарів, як показано на малюнку нижче. У результаті дискримінатор стає функція декількох масштабних виходів генератора (використовуючи операції конкатенації) і, що важливо, передає градієнти одночасно всім масштабам. Приклад роботи MSG-GAN наведений на рисунку 2.14.

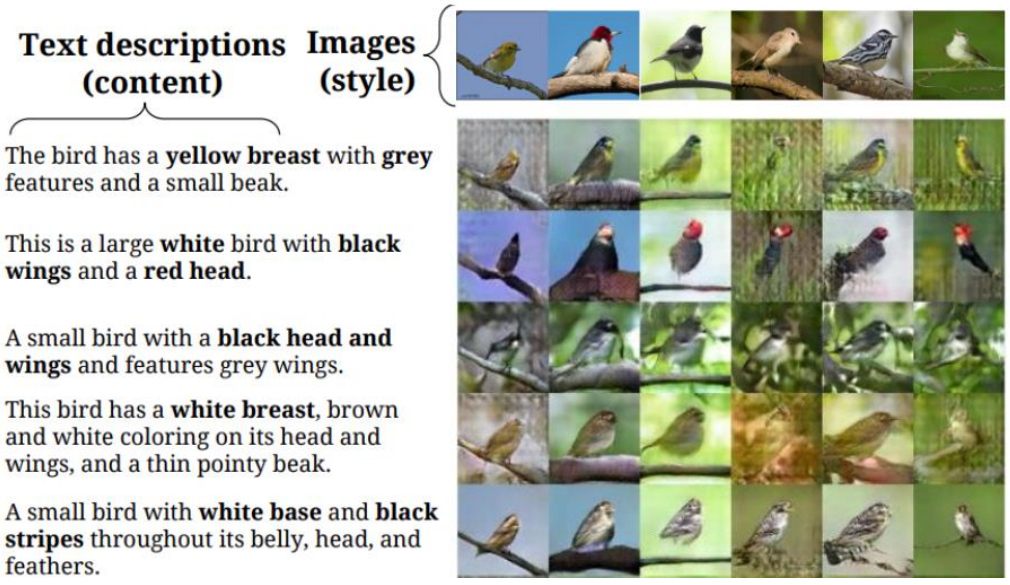

Рисунок 2.14 – Приклад роботи MSG-GAN

Архітектура MSG-GAN для створення синхронізованих багатомасштабних зображень. Пропонована архітектура включає з'єднання від проміжних рівнів генератора до проміжних рівнів дискримінатора. Мультимасштабні зображення, що вводяться в дискримінатор, перетворюються на просторові обсяги, які поєднуються з відповідними активаційними обсягами, отриманими з основного шляху згорткових шарів. MSG-GAN стійкий до змін у швидкості навчання та має більш стійке підвищення якості зображення порівняно з прогресивним зростанням (Pro-GAN). MSG-GAN демонструє однакову характеристику конвергенції та узгодженість для всіх дозволів, а зображення, що генеруються при вищій роздільній здатності, підтримують симетрію певних функцій, таких як той самий колір для обох очей або сережки в обох вухах. Крім того, фаза навчання дозволяє краще зрозуміти властивості зображення (наприклад, якість та різноманітність). Архітектура MSG-GAN наведена на рисунку 2.15.

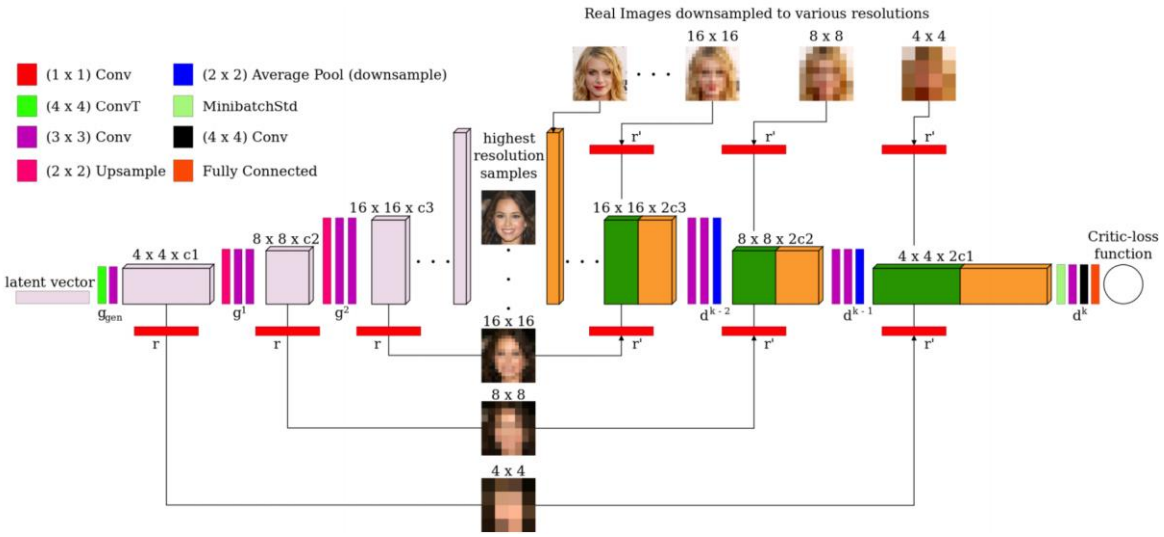

Рисунок 2.15 – Архітектура MSG-GAN

Шари з більш високою роздільною здатністю спочатку відображають блоки однотонних кольорів, але зрештою навчання проникає у всі шари, а

потім всі вони працюють в унісон для отримання кращих зразків. У перші кілька секунд тренування особи, схожі на краплі, з'являються в послідовному порядку від найнижчої роздільної здатності до найвищої роздільної здатності.

Об'єктно-керований синтез тексту зображення через змагальне навчання. Середина: модифікована реалізація двоетапного (layout-image) покоління. Внизу: Obj-GAN та його об'єктно-орієнтована візуалізація уваги. Середнє і нижнє покоління використовують одну й ту саму згенеровану семантичну компоновку, і єдина відмінність полягає в об'єктно-орієнтованій увазі [34]. Приклад об’єктно-керованого синтезу текста наведений на рисунку 2.16.

Рисунок 2.16 – Приклад об’єктно-керованого синтезу текста

Об'єктно-керований уважний GAN (Obj-GAN) виконує дрібнозернистий синтез тексту зображення в два етапи: генерування семантичного макета (мітки класів, що обмежують рамки, форми виступаючих об'єктів), а потім генерування зображення шляхом синтезу

зображення за допомогою генератора згорткових зображень. Генерація семантичної розмітки завершується, коли Obj-GAN приймає пропозицію як вхідні дані і генерує семантичну розмітку, послідовність об'єктів, зазначену їх обмежувальними рамками (з мітками класів) та форми.

На етапі генерації зображення об'єктно-керований уважний генератор також призначений для забезпечення можливості генерації зображення на основі семантичного макета, згенерованого на першому етапі. Генератор концентрується на синтезі області зображення всередині прямокутника, що обмежує, фокусуючись на словах, які найбільш актуальні для об'єкта в цьому обмежувальному прямокутнику. Вектори контексту, що керуються увагою, використовують як контекстні, так і об'єктні вектори контексту для конкретних областей зображення, щоб кодувати інформацію зі слів, які найбільш актуальні для цієї області зображення.

Об'єктно-орієнтована увага (звертаючи увагу на найбільш релевантні слова та попередньо згенеровані мітки класів) працює краще, ніж традиційна увага сітки, здатна створювати складні сцени у високій якості. Принцип роботи Obj-GAN наведений на рисунку 2.17.

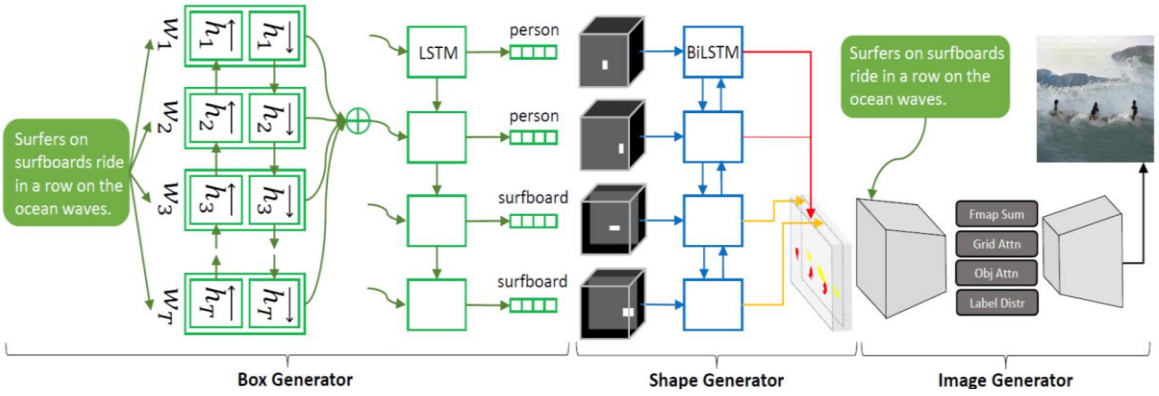

Рисунок 2.17 – Принцип роботи Obj-GAN

StoryGAN: послідовний умовний GAN для візуалізації історії. Візуалізація історії приймає як вхідні дані абзац з декількох речень і

генерує на його виході послідовність зображень, по одному на кожну пропозицію. Завдання візуалізації історії – це послідовне завдання умовної генерації, у якій спільно розглядає поточне вхідне речення з контекстної інформацією. У StoryGAN менше уваги приділяється безперервності зображень (кадрів), що генеруються, але більше уваги до глобальної узгодженості динамічних сцен і персонажів.

Покладається на компонент Text2Gist у Context Encoder, де Context Encoder динамічно відстежує потік сюжету на додаток до забезпечення генератора зображень як локальної, і глобальної умовної інформацією [35]. Двохрівневий дискримінатор та рекурентна структура входів допомагають підвищити якість зображення та забезпечити узгодженість згенерованих зображень та історії, що підлягає візуалізації. Архітектура StoryGAN наведена на рисунку 2.18.

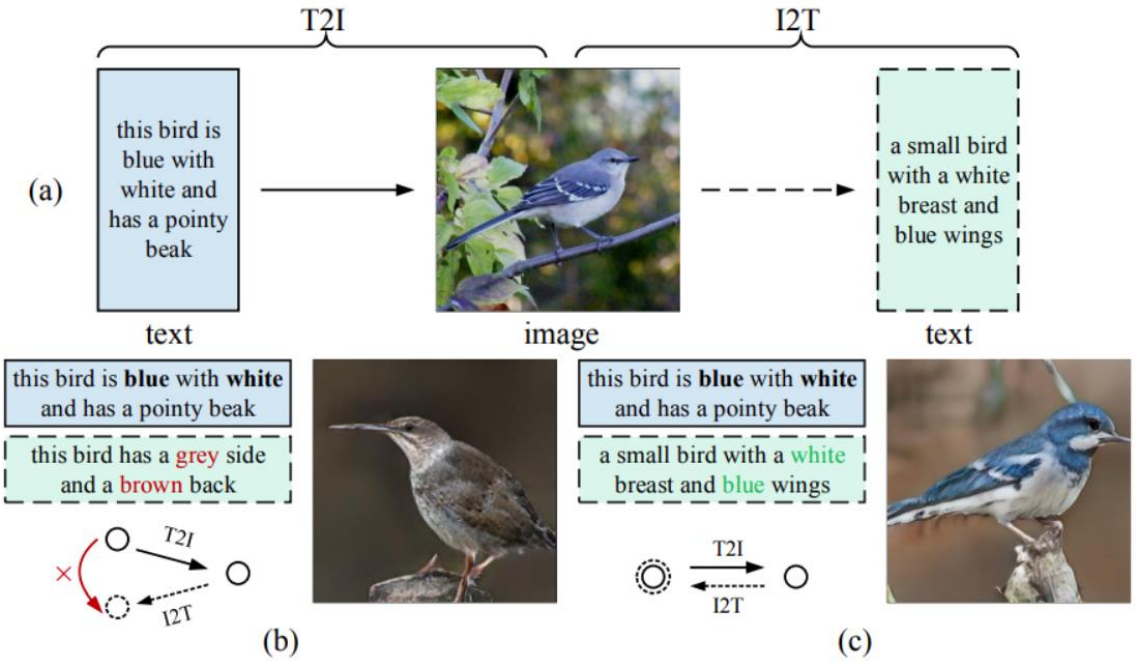

Рисунок 2.18 – Архітектура StoryGAN

Каркас StoryGAN. Архітектура Story GAN здатна розрізняти реальні/підроблені історії за допомогою векторів ознак

зображень/пропозицій в історії, коли вони об'єднуються. Продукт зображення та тексту вводиться у пов'язаний шар із сигмоїдальною нелінійністю, щоб передбачити, чи це фіктивною чи реальною парою історій. Структура історії дискримінатора наведена на рисунку 2.19. Результат генерації шляхом зміни імен персонажів наведений на рисунку 2.20.

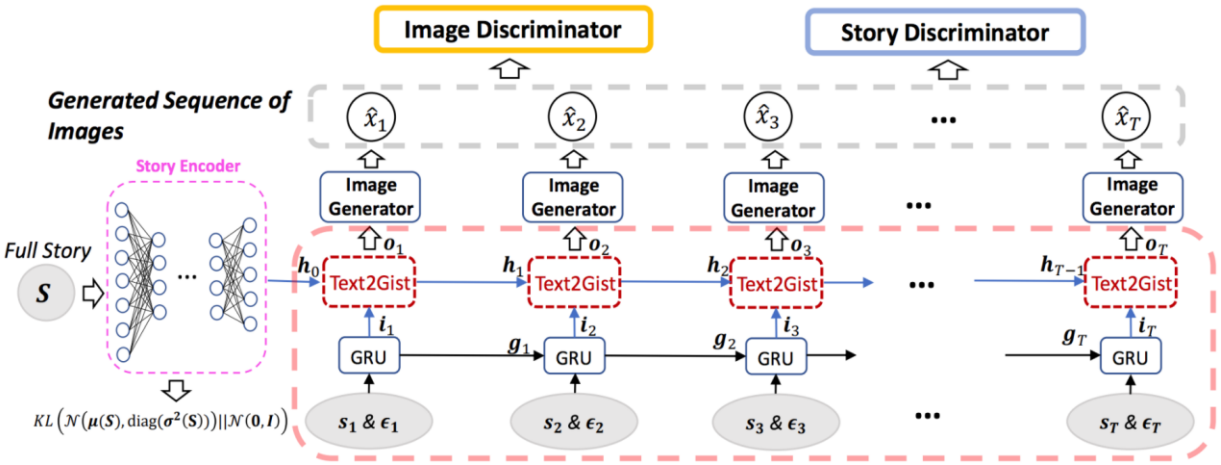

Рисунок 2.19 – Структура історії дискримінатора

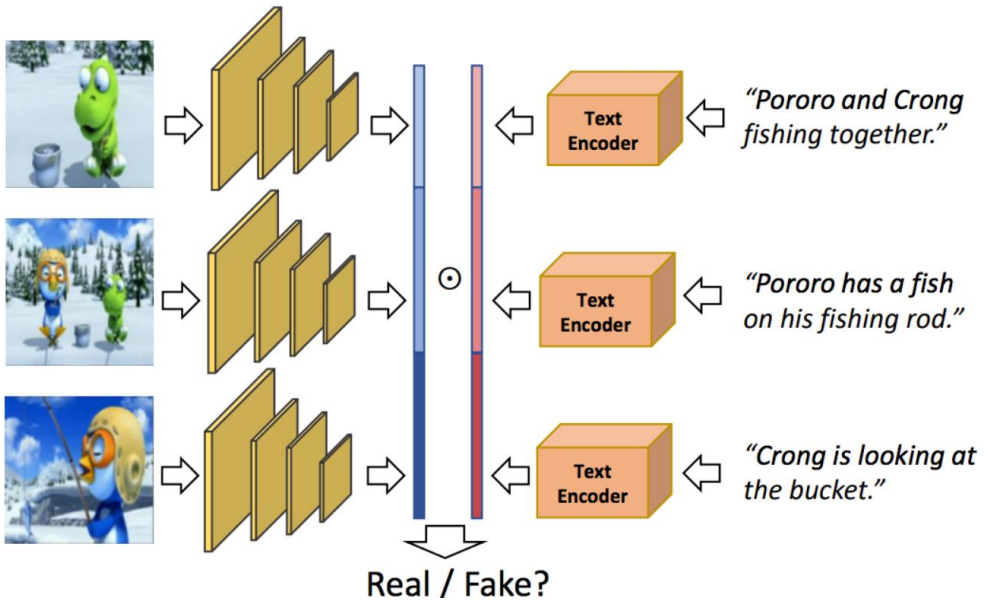

Рисунок 2.20 – Результат генерації шляхом зміни імен персонажів

Keras-текст до зображення. У Keras переклад тексту зображення здійснюється за допомогою GAN і Word2Vec, а також за допомогою періодичних нейронних мереж. Воно використовує DCGan. Приклад архітектури DCGAN для генерації зображень наведений на рисунку 2.21.

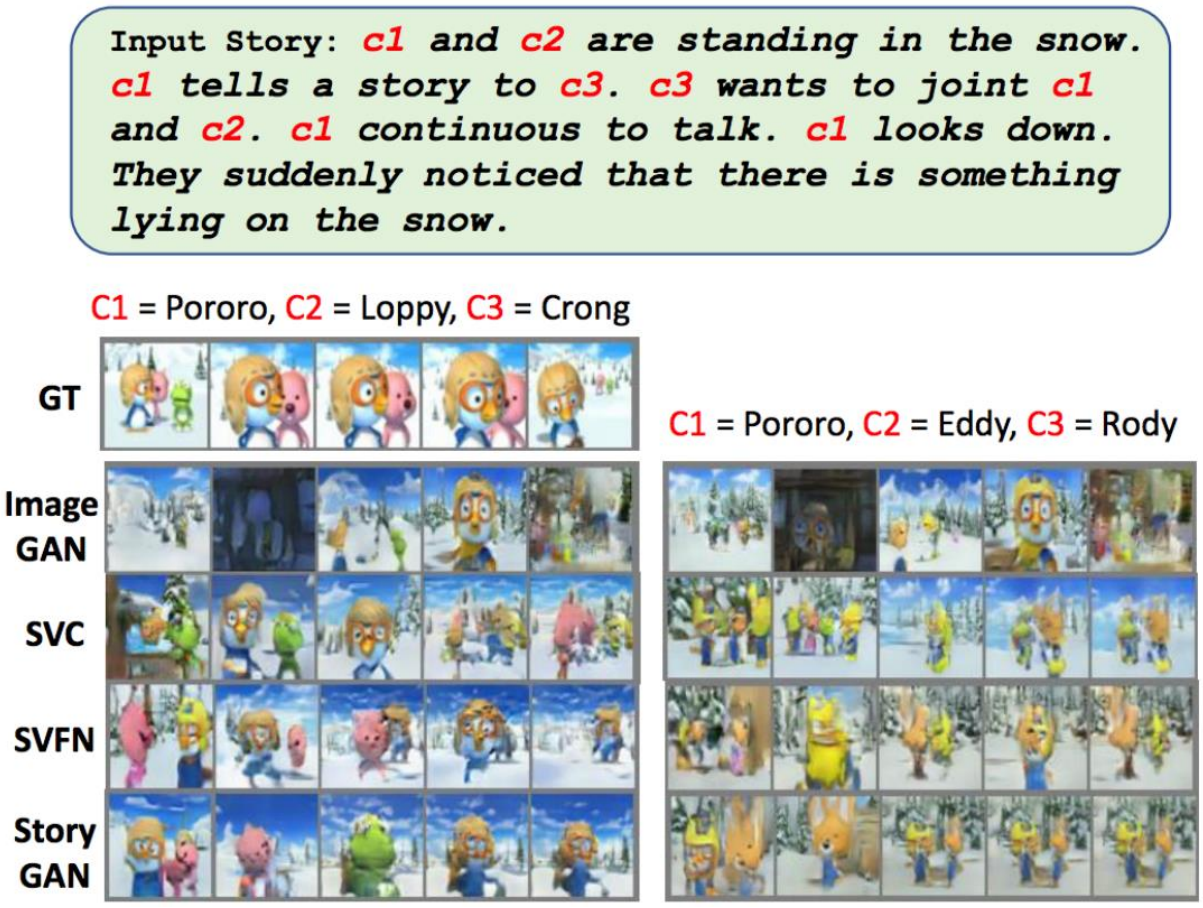

Рисунок 2.21 – Приклад архітектури DCGAN для генерації зображень

Це був прорив у дослідженнях GAN, оскільки він вносить значні зміни в архітектуру для вирішення таких проблем, як нестабільність навчання, колапс режиму та зміна внутрішньої коваріації.

3 ОПИС УВАЖНО ЗМАГАЛЬНОЇ ГЕНЕРАТИВНОЇ МЕРЕЖІ ДЛЯ ВИРІШЕННЯ ЗАДАЧІ ПЕРЕТВОРЕННЯ

У цьому розділі ми розглянемо уважну генеративну змагальну мережу (AttnGAN), яка орієнтована на увагу, багатоступеневе уточнення для генерування дрібнозернистого тексту в зображення. Завдяки новій мережі генерації уваги AttnGAN може синтезувати дрібнозернисті деталі в різних субрегіонах зображення, звертаючи увагу на відповідні слова в описі природної мови. Крім того, пропонується мультимодальна модель подібності глибокої уваги для обчислення дрібнозернистої втрати відповідності зображення-текст для навчання генератора. Запропонований AttnGAN значно перевершує попередній рівень техніки.

3.1 Введення шодо моделі AttnGAN

Автоматичне генерування зображень відповідно до описів природною мовою є фундаментальною проблемою в багатьох програмах, таких як створення мистецтва та комп’ютерне проектування. Це також стимулює прогрес у дослідженні в мультимодальному навчанні та висновках через бачення та мову, що є одним із найактивніших напрямків дослідження останніх років.

Останні запропоновані методи синтезу тексту в зображення засновані на генеративних змагальних мережах (GAN) [9]. Поширеним підходом є кодування всього опису тексту в глобальний вектор речення як умова для створення зображення на основі GAN. Хоча були представлені вражаючі результати, обумовлення GAN лише на глобальному векторі пропозиції не має важливої дрібнозернистої інформації на рівні слів і перешкоджає створенню високоякісних зображень. Ця проблема стає ще більш серйозною під час створення складних сцен, наприклад, у наборі даних COCO.

Щоб вирішити цю проблему, була запропонована Attentional Generative Adversarial Network (AttnGAN), яка дозволяє керувати увагою багатоступеневого уточнення для генерування дрібнозернистого тексту-зображення. Загальна архітектура AttnGAN наведена на рисунку 3.1.

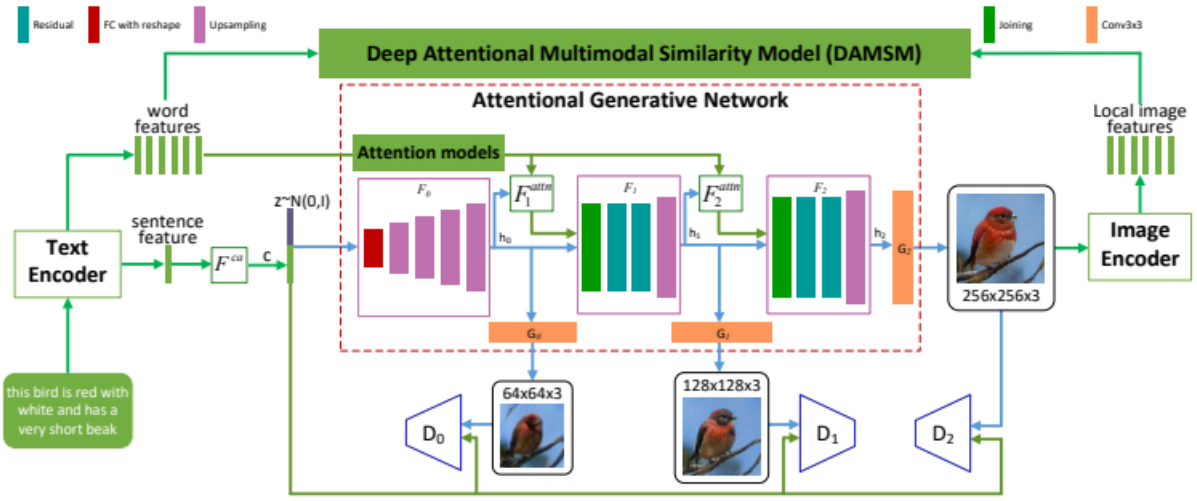

Рисунок 3.1 – Загальна архітектура AttnGAN

Модель складається з двох нових компонентів. Перший компонент – це мережа генерування уваги, в якій розроблено механізм уваги, щоб генератор малював різні підобласті зображення, зосереджуючи увагу на словах, які найбільш релевантні субрегіону, що малюється. Більш конкретно, окрім кодування опису природної мови у глобальний вектор речення, кожне слово в реченні також кодується у вектор слів. Генеративна мережа використовує глобальний вектор речення для створення зображення з низькою роздільною здатністю на першому етапі. На наступних етапах він використовує вектор зображення в кожному субрегіоні для запиту векторів слів за допомогою шару уваги для формування вектора контексту слів. Потім він поєднує регіональний вектор зображення та відповідний вектор контексту слова, щоб утворити мультимодальний вектор контексту, на основі якого модель генерує нові характеристики зображення в навколишніх субрегіонах. Це фактично дає

зображення з більш високою роздільною здатністю з більшою кількістю деталей на кожному етапі. Іншим компонентом AttnGAN є модель мультимодальної схожості глибокої уваги (DAMSM). Завдяки механізму уваги DAMSM може обчислити схожість між згенерованим зображенням і реченням, використовуючи як інформацію про глобальний рівень речення, так і детальну інформацію про рівень слів. Таким чином, DAMSM забезпечує додаткову дрібнозернисту втрату узгодження зображення-текст для навчання генератора. Приклад результатів запропонованого AttnGAN наведений на рисунку 3.2.

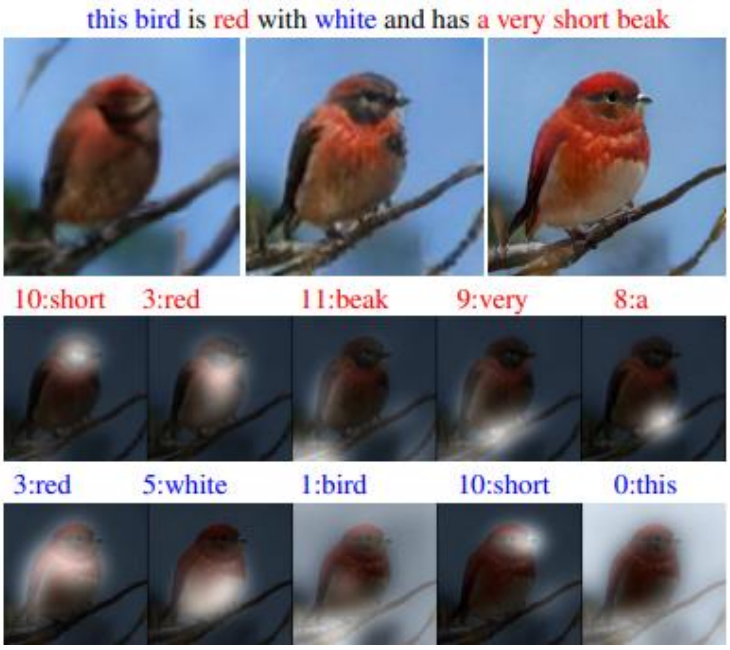

Рисунок 3.2 – Приклад результатів запропонованого AttnGAN

Внесок цього методу потрійний. (i) AttnGAN пропонується для синтезу зображень із текстових описів. Зокрема, в AttnGAN пропонуються два нові компоненти, включаючи генеративну мережу уваги та DAMSM [36]. (ii) Для емпіричної оцінки запропонованого AttnGAN проводиться комплексне дослідження. Результати експерименту показують, що AttnGAN значно перевершує попередні сучасні моделі GAN. (iii) Детальний аналіз виконується шляхом візуалізації шарів уваги AttnGAN.

Вперше показано, що багатошарова умовна GAN здатна автоматично звертатися до відповідних слів, щоб сформувати умову для генерації зображення.

3.2 Опис мережі AttnGAN

Як показано на рисунку 3.1, запропонована генеративна мережа AttnGAN має два нових компоненти: генеративну мережу уваги та мультимодальну модель подібності глибокої уваги. Кожен з них буде розглянуто детальніше в решті частини цього розділу.

3.2.1 Генеративна мережа уваги

Сучасні моделі на основі GAN для генерації тексту в зображення [37], зазвичай кодують текстовий опис цілого речення в один вектор як умову генерації зображення, але не мають детальної інформації на рівні слова. У цьому розділі буде розглянута модель уваги, яка дає змогу генеративної мережі малювати різні субрегіони зображення, обумовлені словами, які є найбільш релевантними для цих субрегіонів.

Як показано на рисунку 3.1, запропонована генеративна мережа уваги має m генераторів ($G_0, G_1, ..., G_{m-1}$), які беруть приховані стани ($h_0, h_1, ..., h_{m-1}$) як вхідні дані та генерують зображення від малого до великого масштабу ($\hat{x}_0, \hat{x}_i, ..., \hat{x}_{m-1}$). Зокрема,

$$h_0 = F_0(z, F^{ca}(\bar{e})), \tag{3.1}$$

$$h_i = F_i\left(h_{i-1}, F_i^{attn}(e, h_{i-1})\right) for\ i = 1, 2, \dots, m-1, \tag{3.2}$$

$$\hat{x}_i = G_i(h_i), \tag{3.3}$$

тут z – вектор шуму, який зазвичай відбирається зі стандартного нормального розподілу. $\bar{e}$ – глобальний вектор речення, а e – матриця векторів слів. F^{ca} представляє збільшення кондиціонування, який перетворює вектор речення $\bar{e}$ у вектор умовлення [38]. F^{attn} є запропонованою моделлю уваги на i^{th} етапі AttnGAN. F^{ca}, F^{attn}, F_i і G_i моделюються як нейронні мережі. Модель уваги $F^{attn}(e,h)$ має вхідні дані: характеристики слова $e \in R^{DxT}$ і ознаки зображення з попереднього прихованого шару $h \in R^{DxN}$. Словні особливості спочатку перетворюються в загальний семантичний простір елементів зображення шляхом додавання нового персептронного шару, тобто $\acute{e} = Ue$, де $U \in R^{DxD}$. Потім для кожної підобласті зображення обчислюється вектор контексту слова на основі його прихованих ознак h (запит). Кожен стовпець h є вектором ознак субрегіону зображення. Для j^{th} субрегіон, його вектор текстового контексту є динамічним представленням векторів слів, що мають відношення до h_j , яке обчислюється за допомогою

$$c_j = \sum_{i=0}^{T-1} \beta_{j,i}\acute{e}_i, where\ \beta_{j,i} = \frac{\exp(\acute{s}_{j,i})}{\sum_{k=0}^{T-1}\exp(\acute{s}_{j,k})}, \tag{3.4}$$

де $\acute{s}_{j,i} = h_j^T \acute{e}_i$, та $\beta_{j,i}$ вказує на вагу моделі до i^{th} слова під час створення j^{th} субрегіону зображення. Потім ми передаємо матрицю слово-контекст для набору функцій зображення h від $F^{attn}(e,h) = (c_0, c_1, \ldots, c_{N-1}) \in R^{DxN}$. Нарешті, функції зображення та відповідні функції контексту слів об’єднуються для створення зображень на наступному етапі. Для створення реалістичних зображень з кількома рівнями (тобто на рівні речення та рівні слова) умов кінцева цільова функція генеративної мережі уваги визначається як

$$\mathcal{L} = \mathcal{L}_G + \lambda \mathcal{L}_{DAMSM}, where\ \mathcal{L}_G = \sum_{i=0}^{m-1} \mathcal{L}_{Gi}, \quad (3.5)$$

тут λ є гіперпараметром для балансування двох членів рівняння. Перший доданок – це втрати GAN, які разом апроксимують умовні та безумовні розподіли [39]. На i^{th} етапі AttnGAN генератор G_i має відповідний дискримінатор D_i. Змагальні втрати для G_i визначають як

$$\mathcal{L}_{Gi} = -\frac{1}{2}\mathrm{E}_{\hat{x}i\sim PGi}[\log(D_i(\hat{x}_i))], \quad (3.6)$$

$$\mathcal{L}_{Gi} = -\frac{1}{2}\mathrm{E}_{\hat{x}i\sim PGi}[\log(D_i(\hat{x}_i, \bar{e}))], \quad (3.7)$$

де безумовна втрата визначає, чи є зображення справжнім чи підробленим, а умовна втрата визначає, чи збігаються зображення та речення чи ні. На відміну від навчання G_i, кожен дискримінатор D_i навчається класифікувати вхідні дані в клас реальних або підроблених шляхом мінімізації перехресних ентропійних втрат, визначених як

$$\mathcal{L}_{Di} = -\frac{1}{2}\mathrm{E}_{xi\sim Pdatai}[\log(D_i(x_i))] - \frac{1}{2}\mathrm{E}_{\hat{x}i\sim PGi}[\log(1 - D_i(\hat{x}_i)], \quad (3.8)$$

$$\mathcal{L}_{Di} = -\frac{1}{2}\mathrm{E}_{xi\sim Pdatai}[\log(D_i(x_{i,}\bar{e}))] - \frac{1}{2}\mathrm{E}_{\hat{x}i\sim PGi}[\log(1 - D_i(\hat{x}_i, \bar{e})], \quad (3.9)$$

де x_i – від істинного розподілу зображення p_{datai} в i^{th}масштабі, а $\hat{x}_i$ – від модельного розподілу p_{Gi} в тому ж масштабі. Дискримінатори AttnGAN структурно не перетинаються, тому їх можна навчати паралельно, і кожен з них фокусується на одній шкалі зображення.

Другий член рівняння, LDAMSM — це дрібнозерниста втрата

відповідності зображення-текст на рівні слова, обчислена DAMSM, яка буде детально розглянута в наступному підрозділі.

3.2.2 Модель мультимодальної схожості глибокої уваги

DAMSM вивчає дві нейронні мережі, які відображають підобласті зображення та слова речення в загальний семантичний простір, таким чином вимірюючи подібність зображення-текст на рівні слова, щоб обчислити дрібні втрати для створення зображення.

Кодер тексту — це двонаправлена довготривала пам’ять (LSTM) [40], яка витягує семантичні вектори з опису тексту. У двонаправленому LSTM кожне слово відповідає двом прихованим станам, по одному для кожного напрямку. Таким чином, ми об’єднуємо два його приховані стани, щоб представити семантичне значення слова. Матриця ознак усіх слова позначається $e \in R^{DxT}$. Його i^{th} стовпець e_i є вектором ознак для i^{th} слова. D — розмірність вектора слів, а T — кількість слів. Тим часом, останні приховані стани двонаправленого LSTM об’єднуються в глобальний вектор речення, позначений $\bar{e} \in RD$.

Кодер зображень — це згортка нейронна мережа (CNN), яка відображає зображення в семантичні вектори. Проміжні рівні CNN вивчають локальні особливості різних субрегіонів зображення, а пізні рівні вивчають глобальні особливості зображення. Точніше, кодер зображень побудований на моделі Inception-v3, попередньо навченій на ImageNet. Спочатку ми змінюємо масштаб вхідного зображення 299×299 пікселів. Потім ми витягуємо локальну матрицю ознак $f \in R^{768x289}$ (змінено з 768×17×17) із шару «mixed_6e» Inception-v3 [41]. Кожен стовпець f є вектором ознак субрегіону зображення. 768 – це розмір вектора локальних ознак, а 289 – кількість субрегіонів на зображенні. Тим часом глобальна особливість вектор $f\ R\ 2048$ витягується з останнього середнього шару об’єднання Inception-v3. Нарешті, ми перетворюємо елементи зображення

в загальний семантичний простір текстових об’єктів, додаючи шар персептрону:

$$v = Wf, \qquad \bar{v} = \overline{Wf}, \tag{3.10}$$

де $v \in R^{Dx289}$ і його i^{th} стовпець v_i – вектор візуальних ознак для i^{th} субрегіону зображення; і $v \in R^D$ – глобальний вектор для всього зображення. D – це розмір мультимодального простору об’єктів (тобто модальностей зображень і текстів). Для ефективності всі параметри в шарах, створених на основі моделі Inception-v3, фіксуються, а параметри в щойно доданих шарах вивчаються спільно з рештою мережі.

Оцінка відповідності зображення-текст, орієнтована на увагу, призначена для вимірювання відповідності пари зображення-речення на основі моделі уваги між зображенням і текстом. Спочатку обчислюємо матрицю подібності для всіх можливих пар слів у реченні та підобластей на зображенні за допомогою

$$s = e^T v, \tag{3.11}$$

де $s \in R^{Tx289}$ і $s_{i,j}$ – подібність добутку між i^{th} слово речення та j^{th} підобласть зображення. Знаходимо, що вигідно нормалізувати матрицю подібності наступним чином

$$\bar{s}_{i,j} = \frac{\exp(s_{i,j})}{\sum_{k=0}^{T-1} \exp(s_{k,j})}. \tag{3.12}$$

Потім ми будуємо модель уваги, щоб обчислити вектор контексту регіону для кожного слова (запиту). Вектор контексту регіону c_i – це динамічне представлення субрегіонів зображення, пов’язаних з i^{th} словом речення. Він обчислюється як зважена сума за всіма регіональними

візуальними векторами, тобто

$$c_i = \sum_{j=0}^{288} a_j v_j, \qquad where\ a_j = \frac{\exp(\gamma_1 \bar{s}_{i,j})}{\sum_{k=0}^{288} \exp(\gamma_1 \bar{s}_{i,k})}, \tag{3.13}$$

тут γ_1 є фактором, який визначає кількість уваги враховуючи особливості відповідних субрегіонів під час обчислень вектору контексту регіону для слова. Нарешті, ми визначаємо релевантність між i^{th} словом і зображенням, використовуючи косинус подібності між c_i та e_i, тобто $R(c_i, e_i) = (c_i^T e_i)/(||c_i||||e_i||)$ [42]. Натхненний формулюванням мінімальної помилки класифікації при розпізнаванні мовлення, оцінка відповідності зображення та тексту, що керується увагою, між цілим зображенням (Q) і всім описом тексту (D) визначається як

$$R(Q, D) = \log(\sum_{i=1}^{T-1} exp(\gamma_2 R(c_i, e_i))) \frac{1}{\gamma_2}, \tag{3.14}$$

де γ_2 – коефіцієнт, який визначає, наскільки збільшити важливість найбільш релевантної пари «слово-регіон». Коли $\gamma_2 \rightarrow \infty, R(Q, D)$ наближається до $max_{i=1}^{T-1} R(c_i, e_i)$.

Втрата DAMSM призначена для вивчення моделі уваги в напівконтрольований спосіб, за якого єдиним контролем є узгодження між цілими зображеннями та цілими реченнями (послідовністю слів). Подібно до [43], для партії пар зображення-речення $\{(Q_i, D_i)\}_{i=1}^{M}$, апостеріорна ймовірність того, що речення D_i збігається із зображенням Q_i обчислюється як

$$P(Di\ |Qi) = \frac{\exp(\gamma_3 R(Q_i, D_i))}{\sum_{j=1}^{M} \exp(\gamma_3 R(Q_i, D_j))}, \tag{3.15}$$

де γ_3 – коефіцієнт згладжування, визначений експериментами. У цій партії речень лише D_i відповідає зображенню Q_i, і розглядати всі інші $M-1$ речення як невідповідні описи. Дотримуючись [43], ми визначаємо функцію втрат як негативний логарифм апостеріорної ймовірності того, що зображення узгоджуються з їх відповідними текстовими описами (основна істина), тобто

$$\mathcal{L}_1^w = -\sum_{i=1}^{M} \log P(D_i \,|Q_i)\,, \tag{3.16}$$

де «*w*» означає «слово». Симетрично ми також мінімізуємо

$$\mathcal{L}_2^w = -\sum_{i=1}^{M} \log P(Q_i \,|D_i)\,, \tag{3.17}$$

де $P(Q_i \,|D_i) = \frac{exp(\gamma_3 R(Q_i,D_i))}{\sum_{j=1}^{M} \exp(\gamma_3 R(Q_j,D_i))}$ – задана ймовірність того, що речення D_i співпадає з відповідним зображенням Q_i. Якщо ми перевизначимо рівняння через $R(Q,D) = (\bar{v}^T \bar{e})/(\left|\left|\bar{v}\right|\right|\left|\left|\bar{e}\right|\right|)$ і підставити його в рівняння (14), (15) і (16), можна отримати функції втрат $\mathcal{L}_1^s$ і $\mathcal{L}_2^s$ (де «s» означає «речення») з використанням вектора пропозиції $\bar{e}$ та глобального вектора зображення $\bar{v}$.

Нарешті, втрата DAMSM визначається як

$$\mathcal{L}_{DAMSM} = \mathcal{L}_1^w + \mathcal{L}_2^w + \mathcal{L}_1^s + \mathcal{L}_2^s. \tag{3.18}$$

На основі експериментів із затриманим набором перевірки ми встановили гіперпараметри в цьому розділі як: $\gamma_1 = 5, \gamma_2 = 5, \gamma_3 = 10$ і $M = 50$. Наш DAMSM попередньо навчений шляхом мінімізації $\mathcal{L}_{DAMSM}$ за допомогою реальних пар зображення-текст. Оскільки розмір зображень

для попереднього навчання DAMSM не обмежений розміром зображень, які можна створити, реальні зображення використовуються розміри 299×299. Крім того, попередньо підготовлений текстовий кодер у DAMSM забезпечує візуально-дискримінаційні вектори слів, засвоєні з парних даних зображення-текст, для генеративної мережі уваги. Для порівняння, звичайні вектори слів, попередньо навчені на чистих текстових даних, часто не є візуально-дискримінаційними, наприклад, вектори слів різних кольорів, таких як червоний, синій, жовтий тощо, часто групуються разом у векторному просторі через відсутність заземлення їх на реальні візуальні сигнали.

Підсумовуючи, пропонуємо дві нові моделі уваги, генеративну мережу уваги та DAMSM, які відіграють різні ролі в AttnGAN. (i) Механізм уваги в генеративній мережі (див. рівняння 4) дозволяє AttnGAN автоматично вибирати умови рівня слова для генерування різних субрегіонів зображення. (ii) Завдяки механізму уваги (див. рівняння 13), DAMSM здатний обчислювати $\mathcal{L}_{DAMSM}$ для дрібнозернистого узгодження тексту-зображення. Варто зазначити, що $\mathcal{L}_{DAMSM}$ застосовується лише на виході останнього генератора G_{m-1}, оскільки кінцевою метою AttnGAN є створення великих зображень останнім генератором. Ми спробували застосувати $\mathcal{L}_{DAMSM}$ до зображень усіх роздільних здатностей, створених $(G_0, G_1, \ldots, G_{m-1})$. Однак продуктивність не була покращена, але зросла вартість обчислень.

4 ПРОГРАМНА РЕАЛІЗАЦІЯ ВЕБ-ДОДАТКУ ТА АНАЛІЗ ОТРИМАНИХ РЕЗУЛЬТАТІВ

4.1 Компоненти системи

4.1.1 Бібліотеки системи

Розроблений додаток призначений для перетворення текстового опису у зображення. З самого початку нашої роботи треба підключити всі необхідні бібліотеки для роботи з перетворенням. Кожна з наших бібліотек відповідає за конкретні функції ті дії нашої програми, та дає всі можливості для роботи з ними.

PyTorch – це фреймворк машинного навчання (ML) з відкритим вихідним кодом, заснований на мові програмування Python і бібліотеці Torch. Це одна з найкращих платформ для досліджень глибокого навчання. Структура створена для прискорення процесу між створенням дослідницького прототипу та розгортанням. PyTorch подібний до NumPy і обчислює за допомогою тензорів, які прискорюються графічними процесорами (GPU). Тензори – це масиви, тип багатовимірної структури даних, з якою можна оперувати та керувати за допомогою API. Фреймворк PyTorch підтримує понад 200 різних математичних операцій. Популярність PyTorch продовжує зростати, оскільки він спрощує створення моделей штучної нейронної мережі (ANN). PyTorch в основному використовується для досліджень, науки про дані та штучного інтелекту (ШІ).

Torchvision – це бібліотека для комп’ютерного зору, яка йде рука об руку з PyTorch. Він має утиліти для ефективного перетворення зображень і відео, деякі часто використовувані попередньо навчені моделі та деякі набори даних. Пакет torchvision складається з популярних наборів даних, архітектур моделей та поширених перетворень зображень для комп’ютерного зору.

Streamlit – це фреймворк Python з відкритим кодом для створення веб-програм для машинного навчання та науки про дані. Ми можемо миттєво розробляти веб-програми та легко розгортати їх за допомогою Streamlit. Streamlit дозволяє вам писати програму так само, як ви пишете код на Python. Streamlit дозволяє легко працювати над інтерактивним циклом кодування та перегляду результатів у веб-додатку.

NumPy, що розшифровується як Numerical Python, – це бібліотека, що складається з багатовимірних об’єктів масиву та набору підпрограм для обробки цих масивів. За допомогою NumPy можна виконувати математичні та логічні операції над масивами. NumPy – це пакет Python. Це розшифровується як «Numerical Python». Це бібліотека, що складається з багатовимірних об’єктів масиву та набору підпрограм для обробки масиву.

Scikit-image – це бібліотека обробки зображень з відкритим вихідним кодом для мови програмування Python. Він включає в себе алгоритми для сегментації, геометричних перетворень, маніпуляцій колірним простором, аналізу, фільтрації, морфології, виявлення ознак тощо.

Бібліотека Pillow забезпечує широку підтримку форматів файлів, ефективне внутрішнє представлення та досить потужні можливості обробки зображень. Основна бібліотека зображень розроблена для швидкого доступу до даних, що зберігаються в кількох основних форматах пікселів. Він повинен забезпечити міцну основу для загального інструменту обробки зображень.

YAML – це формат серіалізації даних, розроблений для читання людиною та взаємодії з мовами сценаріїв. PyYAML – це синтаксичний аналізатор і випромінювач YAML для Python.

Pandas – це програмна бібліотека, написана для мови програмування Python для маніпуляції та аналізу даних. Зокрема, він пропонує структури даних та операції для маніпулювання числовими таблицями та часовими рядами.

Набір інструментів природної мови (NLTK) – це платформа, що використовується для створення програм Python, які працюють з даними людської мови для застосування в статистичній обробці природної мови (NLP). Він містить бібліотеки для обробки тексту для токенізації, синтаксичного аналізу, класифікації, створення основ, тегів і семантичного міркування.

4.2 Функції розроблюваної системи

З самого початку повинно встановити усі потрібні для нашої роботи бібліотеки. Варто помітити, що у кожної бібліотеки є своя версія, яка не завжди працює с іншими версіями бібліотек. Для кожної версії бібліотеки рекомендується спочатку прочитати опис. Зі списком бібліотек, які використовуються у проекті можна ознайомитися у підрозділі 4.1.

Далі, ми загружаємо і працюємо з вже потренованими DAMSMencoders, а саме з такими файлами, як image_encoder та text_encoder. Як було описано у розділі 3.2.2, механізм уваги обчислює подібність між згенерованим зображенням і реченням, використовуючи глобальну інформацію про рівень речення та детальну інформацію про рівень слів. Тобто, механізм уваги забезпечує додаткову дрібнозернисту втрату узгодження зображення-текст для навчання генератора. Файл text_encoder представляє з себе двонаправлену довготривалу пам’ять (LSTM), яка витягує семантичні вектори з опису тексту. У цій пам’яті кожне слово відповідає двом прихованим станам, по одному для кожного напрямку. Таким чином, ми об’єднуємо два його приховані стани, щоб представити семантичне значення слова. Файл image_encoder представляє з себе згортку нейронної мережі (CNN), яка відображає зображення в семантичні вектори. Проміжні рівні CNN вивчають локальні особливості різних субрегіонів зображення, а пізні рівні вивчають глобальні особливості зображення.

В якості даних про птахів використовується датасет CUB_200_2011. Caltech-UCSD Birds-200-2011 (CUB-200-2011), який є розширеною версією набору даних CUB-200 з приблизно подвоєною кількістю зображень на клас і новими анотаціями розташування деталей. Набір даних включає в себе 200 категорій птахів, файл classes.txt (наприклад, чорноногий альбатрос, лейсан альбатрос); 11788 зображень, файл images.txt, де кожен рядок відповідає одному зображенню; 15 місць розташування деталей (наприклад, спина, голова); 321 бінарних атрибутів та файл bounding_boxes.txt, в якому кожне зображення містить одну мітку обмежувальної рамки.

DAMSM loss, що є критерієм оцінки втрати якості зображення. За допомогою цієї втрати ми відповідаємо на питання, чи дійсно зображення відповідає текстовому опису? Також, отримаємо бал для пари зображення-текст.

Для подальшої роботи зробити обчислення. Глобальна увага використовує матрицю та метрику запиту. На основі кожного вектора запиту q він обчислює параметризовану опуклу комбінацію на основі матриці.

Після цього в файлі main.py відбувається тренування наших даних, а далі – сам процес генерації текстового опису в зображення.

В файлі model.py відбувається згортка з прокладкою, спочатку 1x1, потім 3x3. Далі йде процес encoder-decoder, де ми визначаємо розмір, вектор вбудовування, кількість наших шарів, кількість прихованих шарів.

В файлі trainer.py виконується завдання до задачі перетворення текстового опису в зображення, підключення файлів text_encoder та image_encoder, описана робота генераторів та дискримінаторів, підготування навчальних даних та обчислення вбудовування тексту, генерація фейкових зображень, обновлення D нейронної мережі, обновлення G нейронної мережі.

За допомогою бібліотеки streamlit був створений веб-інтерфейс

нашого додатку. Для відображення та запуску веб-інтерфейсу нам потрібна команда *python -m streamlit run app.py*, після вводу цієї команди в терміналі python, системи відобразить URL для входу у додаток, де користувач буде мати змогу ввести текстовий опис птаха, який його цікавить. Інтерфейс веб-додатку наведений на рисунку 4.1.

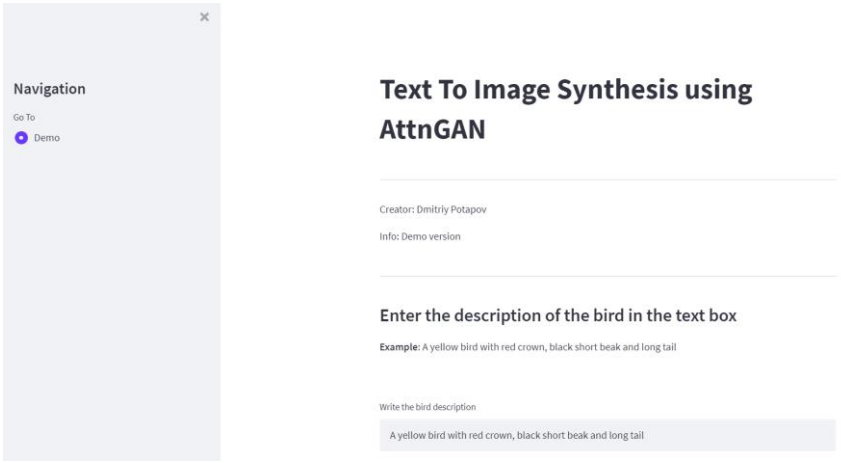

Рисунок 4.1 – Інтерфейс веб-додатку

Далі йде наша модель AttGAN2, в папці якої формуються три зображення після того, як ми ввели опис птаха у текстове поле в нашому веб-додатку. Перша фотографія формату 64x64x3 є фотографією після проходження генератора_0, на якій відображаються висота, ширина, початкові кольори птахів. Перша згенерована фотографія формату 64x64x3 наведена на рисунку 4.2.

Рисунок 4.2 – Перша згенерована фотографія формату 64x64x3

Друга фотографія формату 128x128x3 є фотографією після проходження генератора_1, на якій висота, ширина не змінюються, а забираються зайві забруднення. Друга згенерована фотографія формату 128x128x3 наведена на рисунку 4.3.

Рисунок 4.3 – Друга згенерована фотографія формату 128x128x3

Третя фотографія формату 256x256x3 є фінальною фотографією після проходження генератора2, на якій відображається чітка картинка. Третя згенерована фотографія формату 256x256x3 наведена на рисунку 4.4.

Рисунок 4.4 – Третя згенерована фотографія формату 256x256x3

4.3 Аналіз отриманих результатів

Для оцінки запропонованого AttnGAN проводяться широкі експерименти. Спочатку ми вивчаємо важливі компоненти AttnGAN, включаючи генеративну мережу уваги та DAMSM. Потім ми порівнюємо наш AttnGAN з попередніми найсучаснішими моделями GAN для синтезу тексту в зображення.

Окремої уваги заслуговує характеристика наборів даних. Як і попередні методи перетворення тексту в зображення, цей метод оцінюється на наборах даних CUB та COCO. Ми попередньо обробляємо набір даних CUB відповідно до методу. На рисунку 4.5 наведено статистику наборів даних.

Dataset	CUB		COCO	
	train	test	train	test
#samples	8,855	2,933	80k	40k
caption/image	10	10	5	5

Рисунок 4.5 – Статистика датасетів

В роботі використовується початковий бал як кількісний показник оцінки. Оскільки початковий бал не може відображати, чи добре згенероване зображення обумовлене даним текстовим описом, пропонується використовувати *R*-точність, загальну метрику оцінки для ранжування результатів пошуку, як додаткову метрику оцінки для завдання синтезу тексту в зображення. Якщо є R відповідних документів для запиту, ми розглядаємо результати пошуку з найвищим рейтингом у системі і знаходимо, що r є релевантними, а потім за визначенням *R*-точність дорівнює r/R. Точніше, ми проводимо експеримент пошуку, тобто використовуємо згенеровані зображення для запиту відповідних текстових описів. По-перше, кодери зображень і тексту, засвоєні в

попередньо підготовленому DAMSM, використовуються для вилучення глобальної функції векторів створених зображень і наведені текстові описи. Потім обчислюється косинус подібності між глобальними векторами зображення та глобальними векторами тексту. Нарешті, оцінюються кандидатні текстові описи для кожного зображення за спадною схожістю та знаходяться верхні r відповідних описів для обчислення R-точності. Щоб обчислити початковий бал і R-точність, кожна модель генерує 30 000 зображень із випадково вибраних невидимих текстових описів. Описи тексту-кандидата для кожного зображення запиту складаються з однієї основної істини (тобто $R = 1$) і 99 випадково вибраних описів, що не збігаються. Початкові оцінки та показники точності R наведені на рисунку 4.6.

Далі, йде кількісна оцінка AttnGAN та його варіантів. Початкові оцінки та показники точності R від нашого AttnGAN та його варіанти в різні епохи на тестових наборах CUB (зверху) і COCO (знизу) наведені на рисунку 4.7.

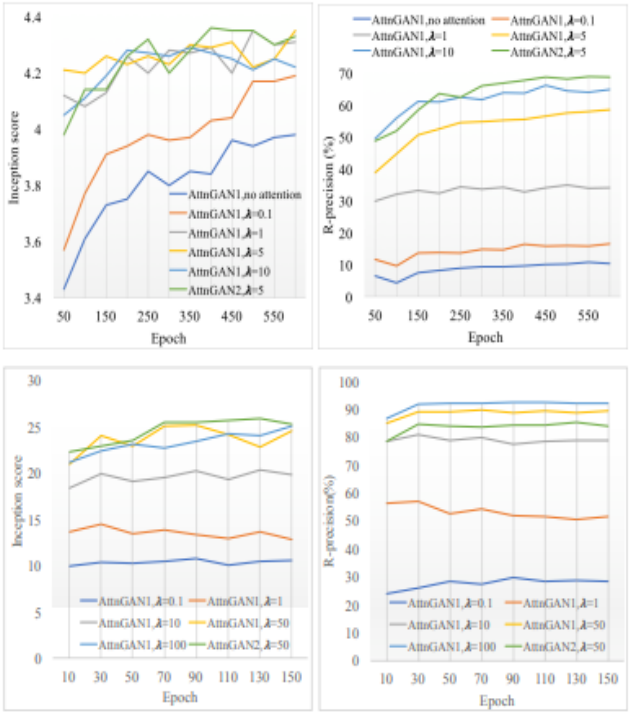

Рисунок 4.6 – Початкові оцінки та показники точності R

Method	inception score	R-precision(%)
AttnGAN1, no attention	$3.98 \pm .04$	10.37 ± 5.88
AttnGAN1, $\lambda = 0.1$	$4.19 \pm .06$	16.55 ± 4.83
AttnGAN1, $\lambda = 1$	$4.35 \pm .05$	34.96 ± 4.02
AttnGAN1, $\lambda = 5$	$4.35 \pm .04$	58.65 ± 5.41
AttnGAN1, $\lambda = 10$	$4.29 \pm .05$	63.87 ± 4.85
AttnGAN2, $\lambda = 5$	**$4.36 \pm .03$**	**67.82 ± 4.43**
AttnGAN2, $\lambda = 50$ (COCO)	**$25.89 \pm .47$**	**85.47 ± 3.69**

Рисунок 4.7 – Найкращий початковий бал і відповідний показник точності R для кожної моделі

Щоб краще зрозуміти, чого дізнався AttnGAN, візуалізуємо його проміжні результати із використанням механізму уваги. Як показано на рисунку 4.8, перший етап AttnGAN (G0) просто створює ескізи примітивної форми та кольорів об’єктів і створює зображення з низькою роздільною здатністю. Оскільки на цьому етапі використовуються лише глобальні вектори речень, у згенерованих зображеннях відсутні деталі, описані точними словами, наприклад, дзьоб та очі птаха. На основі векторів слів наступні етапи (G1 і G2) навчаються виправляти дефекти в результатах попереднього етапу та додавати більше деталей для створення зображень з більш високою роздільною здатністю. Деякі субрегіони/пікселі зображень G1 або G2 можна вивести безпосередньо із зображень, створених на попередньому етапі. Для цих субрегіонів увага рівномірно розподіляється на всі слова і відображається чорним кольором на карті уваги (див. рисунок 4.8). Для інших субрегіонів, які зазвичай мають семантичне значення, виражене в описі тексту, наприклад атрибути предметів, увага приділяється їх найбільш релевантним словам (яскраві області на рисунку 4.8). Таким чином, ці регіони випливають як із особливостей контексту слів, так і з попередніх ознак зображення цих регіонів. Як показано на рисунку 4.8, у наборі даних CUB слова the, this,

bird зазвичай супроводжуються моделями F_{attn} для визначення місця розташування об’єкта; слова, що описують атрибути об'єкта, такі як кольори та частини птахів, також використовуються для виправлення дефектів і деталей малювання. Проміжні результати AttnGAN на тестових наборах CUB і COCO наведені на рисунку 4.8.

Рисунок 4.8 – Проміжні результати AttnGAN на тестових наборах CUB і COCO

У наборі даних COCO маємо подібні спостереження. Оскільки на кожному зображенні COCO зазвичай є більше одного об’єкта, то більш помітно, що слова, що описують різні об’єкти, супроводжуються різними субрегіонами зображення, наприклад, банани, ківі в нижньому правому

блоці рисунка 4.8. Ці спостереження демонструють що AttnGAN вчиться розуміти детальне семантичне значення, виражене в текстовому описі зображення. Інше спостереження полягає в тому, що друга модель уваги F_{attn2} здатна звертати увагу на деякі нові слова, які були опущені першою моделлю уваги F_{attn1} (див. рисунок 4.8). Це демонструє, що для надання більшої інформації для створення зображень з високою роздільною здатністю на останніх етапах AttnGAN, відповідні моделі уваги навчаються відновлювати об’єкти та атрибути, пропущені на попередніх етапах.

Експериментальні результати, наведені вище, кількісно та якісно показали здатність AttnGAN до узагальнення шляхом створення зображень із наведених текстових описів. Тут ми додатково перевіряємо, наскільки чутливі вихідні дані до змін у вхідних реченнях, змінюючи деякі найбільш відвідувані слова в текстових описах. Деякі приклади показано на рисунку 4.9. Він ілюструє, що згенеровані зображення модифікуються відповідно до змін у вхідних реченнях, показуючи, що модель може вловити тонкі семантичні відмінності в описі тексту. Приклади результатів AttnGAN змінюючи найбільш відвідувані слова в описах наведені на рисунку 4.9.

Рисунок 4.9 – Приклади результатів AttnGAN змінюючи найбільш відвідувані слова в описах

З іншого боку, ми також помічаємо, що AttnGAN іноді генерує чіткі та деталізовані зображення, але навряд чи реалістичні. Як приклади, наведені на рисунку 4.10, AttnGAN створює птахів із кількома головами, очима чи хвостами, які існують лише у казках.

Рисунок 4.10 – Нові зображення AttnGAN

Ми можемо сказати, що це вказує на те, що нинішній метод все ще не є досконалим у фіксації глобальних когерентних структур, які залишає простір для вдосконалення.

ВИСНОВКИ

Дана магістерська кваліфікаційна робота присвячена дослідженню нейромережевих методів зіставлення зображень з їх текстовим описом. Під час дослідження стало очевидним, що ця область набула особливої уваги та популярності у роботах, присвячених розробці нових підходів у галузі штучного інтелекту та глибинного навчання. Можно сказати, що перетворення тексту до зображення – це основна проблема в цій галузі, яка також привернула увагу та дослідження багатьох вчених. Постановка задачі перетворення тексту до зображення передбачає створення реалістичного зображення, яке відповідає заданому текстовому опису, що вимагає обробки нечіткої та неповної інформації в описах природною мовою.

Перетворення тексту до зображення стимулює розвиток мультимодального навчання та крос-модальної генерації і демонструє великий потенціал у таких додатках, як крос-модальний пошук інформації, редагування фотографій та автоматизований дизайн. Також генерація зображень закриває важливі потреби сучасного бізнесу – можливість отримати унікальну картинку під власний опис, а також будь-якої миті створювати необхідну кількість licence-free-ілюстрацій. Деякі інші галузі, такі як дизайн одягу, дизайн інтер’єру тощо, є областями застосування синтезу тексту в зображення. Зробити передові медичні зображення або супутникові знімки необхідної якості дуже складно через складність. Синтез тексту в зображення є благословенням для таких галузей. Але, отримати зображення необхідної роздільної здатності та масштабу – це далека мрія.

Після дослідження даної теми можемо сказати, що ця галузь все ще стоїть перед кількома проблемами, які потребують подальших дослідницьких зусиль, таких як можливість створення зображень високої роздільної здатності з кількома об’єктами, і розробка відповідних і

надійних показників оцінки, які корелюють з людськими судженнями. А саме, існує три великі проблеми, такі як: виклик залежності, концептуально-об’єктні відносини та відношення об’єкт-об’єкт. В першому випадку, моделі TTI сильно залежать як від текстових, так і від візуальних методів аналізу, які, хоча вони й досягли значного прогресу в останні роки, мають багато роботи, щоб домогтися масового впровадження. З цього погляду можливості моделей TTI, як правило, обмежені специфікою базового аналізу тексту та моделей генерації зображень. В другому випадку, неймовірно важка проблема, яка має бути вирішена у моделях TTI, – це відносини між концепцією, витягнутою з текстового опису та відповідними візуальними об'єктами. Насправді це може бути безліч об'єктів, відповідних конкретному текстовому опису. З'ясування правильної відповідності між концепціями та об'єктами залишається ключовою проблемою у моделях TTI. В третьому випадку, будь-яке зображення виражає відносини між об'єктами у візуальному форматі. Щоб відобразити це, модель TTI повинна була не тільки генерувати правильні об'єкти, а й відносини між ними. Створення складніших сцен, що містять кілька об'єктів з семантично значущими зв'язками між цими об'єктами, залишається серйозною проблемою технології генерації тексту в зображення.

Далі, був проведений огляд щодо існуючих методів перетворення текстового опису в зображення. Визначились з класифікацією методів: фундаментальні, прямі, текст до зображення з додатковим наглядом та інші. Кожен з описаних методів має свою індивідуальну архітектуру та може похвалитися своїм функціоналом.

Для вирішення задачі перетворення текстового опису у зображення була вибрана уважна генеративна змагальна мережа (AttnGAN), яка орієнтована на увагу, багатоступеневе уточнення для генерування дрібнозернистого тексту в зображення. За допомогою даної мережі генерації уваги може синтезувати дрібнозернисті деталі в різних

субрегіонах зображення, звертаючи увагу на відповідні слова в описі природної мови. Також, пропонується мультимодальна модель подібності глибокої уваги для обчислення дрібнозернистої втрати відповідності зображення-текст для навчання генератора. Запропонована мережа значно перевершує попередній рівень техніки.

Під час розробки програмної реалізації був розроблений додаток, який призначений для перетворення текстового опису у зображення. Інтерфейс додатку дуже простий, так як додаток є демо-версією для демонстрації отриманих результатів.

Після цього був проведений аналіз результатів, де було продемонстровано наш AttnGAN, який значно перевершує найсучасніші моделі GAN, підвищуючи найкращий зареєстрований початковий показник на 14,14% у наборі даних CUB і на 170,25% у складнішому наборі даних COCO. Експериментальні результати демонструють ефективність запропонованих механізмів уваги в AttnGAN, що особливо важливо для генерації тексту в зображення для складних сцен.

Результати роботи відображені у тезах доповіді [44].

ПЕРЕЛІК ДЖЕРЕЛ ПОСИЛАННЯ

1. Xu T. Генерация изображений из текста. URL: https://habr.com/ru/post/424957/ (дата звернення 20.03.2022)

2. Reed S., Akata Z., Mohan S., Tenka S., Schiele B., and Lee H. Learning what and where to draw. URL: https://proceedings.neurips.cc/paper/2016/file/a8f15eda80c50adb0e71943adc8015cf-Paper.pdf (дата звернення 20.03.2021)

3. Reed S., Akata Z., Yan X., Logeswaran L., Schiele B., and Lee H. Generative adversarial text-to-image synthesis. URL: https://proceedings.mlr.press/v48/reed16.html (дата звернення 20.03.2022)

4. Reed S., Scott, et al. Generative adversarial text to image synthesis. URL: https://medium.datadriveninvestor.com/text-to-image-synthesis-6e5de1bf86ec (дата звернення 21.03.2022)

5. Karras T., Aila T., Laine S., Lehtinen J. Progressive growing of gans for improved quality, stability, and variation, in: International Conference on Learning Representations, 2018.

6. Ledig C., Theis L., Husz´ar F., Caballero J. A., Aitken A., Tejani A., Totz J., Wang Z., Shi W. Photo-realistic single image super-resolution using a generative adversarial network, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2016, pp. 4681–4690

7. Athalye A. Neural Style. https://github.com/anishathalye/neural-style.commit (дата звернення 21.03.2022)

8. Yeh R. A., Chen C., Lim T. Y., Schwing A. G., Hasegawa J., M., Do M. N. Semantic image in painting with deep generative models, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2016, pp. 5485–5493.

9. Goodfellow I. J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A. C., and Bengio Y. Generative adversarial nets. URL: https://arxiv.org/pdf/1711.10485.pdf (дата звернення 22.03.2022)

10. Димитров Д. ruDALL-E: генерируем изображения по текстовому описанию с помощью созданной нейронной сети. URL:https://habr.com/ru/company/sberbank/blog/586926/ (дата звернення 22.03.2022)

11. Yu J., Lin Z., Yang J., Shen X., Lu X., Huang T. S. Free-form image in painting with gated convolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 4471–4480.

12. Zhao S. Y., and Li J. W. Generative adversarial network for generating. URL: www.hindawi.com/journals/ddns/2020/6452536/ (дата звернення 23.03.2022)

13. Isola P., Zhu J. Y., Zhou T., Efros A. A. Image-to-image translation with conditional adversarial networks, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2016, pp. 1125–1134.

14. Mirza M., Osindero S. Conditional generative adversarial nets. arXiv:1411.1784 (2014).

15. Reed S. E., Akata Z., Yan X., Logeswaran L., Schiele B., Lee H. Generative adversarial text to image synthesis, in: International Conference on Machine Learning, 2016, pp. 1060– 1069.

16. Nilsback M. E., Zisserman A. Automated flower classification over a large number of classes, in: Indian Conference on Computer Vision, Graphics & Image Processing, 2008, pp. 722–729.

17. Wah C., Branson S., Welinder P., Perona P., Belongie S. The caltech-ucsd birds-200-2011 dataset, California Institute of Technology (2011).

18. Wu K., Xu K., Hall P. A survey of image synthesis and editing with generative adversarial networks, Tsinghua Science and Technology 22 (6) (2017) 660–674.

19. Creswell A., White T., Dumoulin V., Arulkumaran K., Sengupta B., Bharath A. A. Generative adversarial networks: An overview, IEEE Signal Processing Magazine 35 (1) (2018) 53– 65.

20. Wang L., Chen W., Yang W., Bi F., Yu F., R. A state-of-the-art

review on image synthesis with generative adversarial networks, IEEE Access 8 (2020) 63514–63537.

21. Agnese J., Herrera J., Tao H., Zhu X. A survey and taxonomy of adversarial neural networks for text-to-image synthesis, Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery (2020).

22. Zhang H., Xu T., Li H. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks, in: Proceedings of the IEEE International Conference on Computer Vision, 2016, pp. 5907–5915.

23. Ghosh A., Kulharia V., Namboodiri V. P., Philip H.S. Torr, and Puneet K. Dokania. Multi-agent diverse generative adversarial networks. URL: https://openaccess.thecvf.com/content_CVPR_2020/papers (дата звернення 24.03.2022)

24. Zhu M., Pan P., Chen W., Yandg Y. Dm-gan: Dynamic memory generative adversarial networks for text-to-image synthesis, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2019, pp. 5802–5810.

25. Stap D., Bleeker M., Ibrahimi S., ter Hoeve M. Conditional image generation and manipulation for user-specified content, in: Proceedings of the IEEE Computer Vision and Pattern Recognition Workshop, 2020.

26. Yuan M., Peng Y. Bridge-gan: Interpretable representation learning for text-to-image synthesis, IEEE Transactions on Circuits and Systems for Video Technology (2019) 1–1.

27. Joseph K. J., Pal A., Rajanala S., Balasubramanian V., N. C4synth: Cross-caption cycle-consistent text-to-image synthesis, in: IEEE Winter Conference on Applications of Computer Vision, 2018, pp. 358–366.

28. Cheng J., Wu F., Tian Y., Wang L., Tao D. Rifegan: Rich feature generation for text-to-image synthesis from prior knowledge, in: Proceedings of the IEEE Computer Vision and Pattern Recognition, 2020, pp. 10911–10920.

29. Reed S. E., Akata Z., Mohan S., Tenka S., Schiele B., Lee H. Learning what and where to draw, in: Advances in Neural Information Processing Systems, 2016, pp. 217–225.

30. Pennington F., Socher R., Manning C. D. Glove: Global vectors for word representation, in: Proceedings of the Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1532—-1543

31. Krishna R., Zhu Y., Groth O., Johnson J., Hata K., Kravitz J., Chen S., Kalantidis Y., Li L. J., Shamma D. A., et al. Visual genome: Connecting language and vision using crowdsourced dense image annotations, International Journal of Computer Vision 123 (1) (2017) 32–73.

32. Koh J. Y., Baldridge J., Lee H., Yang Y. Text-to-image generation grounded by fine-grained user attention, arXiv:2011.03775 (2020).

33. Голованов В. T2F: проект преобразования текста в рисунок лица при помощи глубинного обучения. URL: https://habr.com/ru/post/420709/ (дата звернення 24.03.2022)

34. Ledig C., Theis L., Huszar F., Caballero J., Aitken A., Tejani A., Totz J., Wang Z., and Shi W. Photo-realistic single image superresolution using a generative adversarial network. URL: https://openaccess.thecvf.com/content_CVPR_2020/papers (дата звернення 25.03.2022)

35. Hong S., Yang D., Choi J., and Lee H. Inferring semantic layout for hierarchical text-to-image synthesis. URL: https://openaccess.thecvf.com/content_CVPR_2020/papers (дата звернення 25.03.2022)

36. Mansimov E., Parisotto E., Ba L. J., and Salakhutdinov R. Generating images from captions with attention. In ICLR, 2016

37. Reed S., Akata Z., Yan X., Logeswaran L., Schiele B., and Lee H. Generative adversarial text-to-image synthesis. In ICML, 2016.

38. Zhang H., Xu T., Li H., Zhang S., Wang X., Huang X., and Metaxas D. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In ICCV, 2017.

39. Zhang H., Xu T., Li H., Zhang S., Wang X., Huang X., and Metaxas D. N. Stackgan++: Realistic image synthesis with stacked generative

adversarial networks. arXiv: 1710.10916, 2017.

40. Schuster M. and Paliwal K. K. Bidirectional recurrent neural networks. IEEE Trans. Signal Processing, 45(11):2673–2681, 1997.

41. Szegedy C., Vanhoucke V., Ioffe S., Shlens J., and Wojna Z. Rethinking the inception architecture for computer vision. In CVPR, 2016.

42. Juang B. H., Chou W., and Lee C. H. Minimum classification error rate methods for speech recognition. IEEE Transactions on Speech and Audio Processing, 5(3):257–265, 1997.

43. Fang H., Gupta S., Iandola F. N., Srivastava R. K., Deng L., Dollar P., Gao J., He X., Mitchell M., Platt J. C., Zitnick C. L., and Zweig G. From captions to visual concepts and back. In CVPR, 2015.

44. Рябова Н.В., Потапов Д.С. Дослідження нейромережевих методів зіставлення зображень та їх текстових анотацій. Матеріали XII Міжнародної науково-технічної конференції «Сучасні напрями розвитку інформаційно-комунікаційних технологій та засобів управління». Баку-Харків-Жиліна, 27-28 квітня 2022.