Харківський національний університет імені В.Н. Каразіна

Факультет математики і інформатики

Кафедра прикладної математики

Кваліфікаційна робота

бакалавра

на тему *«Задача генерації тексту в зображення за допомогою нейронних мереж»*

Виконав:
студент групи МП 41
4 курсу
спеціальність 113 – прикладна математика
освітньо-професійна програма
«Прикладна математика»

Петров М. Ю.

Науковий керівник: *викладач кафедри прикладної математики* ***Карєва В. В.***

Рецензент: *доктор філософії за спеціальністю 111* «*Математика*», *постдок за стипендією Марії Кюрі в Університеті Осло, старший науковий співробітник Фізико-технічного інституту низьких температур ім. Б.І. Вєркіна НАН України* ***Рибалко Я. В.***

Харків - 2025 рік

Анотації

Петров Микола Юрійович. Задача генерації тексту в зображення за допомогою нейронних мереж.

У кваліфікаційній роботі досліджено задачу генерації зображень на основі текстового опису з використанням сучасних глибинних нейронних мереж. Розглянуто основні підходи до генерації, зокрема моделі типу GAN, VAE та дифузійні моделі, з акцентом на архітектурі Stable Diffusion – латентній дифузійній моделі, яка поєднує компоненти CLIP, U-Net та VAE-декодувальник. Особливу увагу приділено методам fine-tuning, таким як Textual Inversion і DreamBooth, що дозволяють адаптувати попередньо натреновані моделі до генерації нових об’єктів за обмеженої кількості навчальних прикладів. У рамках практичної частини реалізовано донавчання моделі Stable Diffusion v1.5 на основі власного датасету, який складається з 15 зображень цільового об’єкта – м’якої іграшки – із відповідними текстовими описами. Експериментальні результати підтвердили ефективність персоналізованої генерації: модель здатна відтворювати цільовий об’єкт у різних стилях, сценах та умовах, зберігаючи його характерні візуальні риси, що засвідчує перспективність такого підходу для подальшого застосування в креативних і прикладних задачах комп’ютерного зору.

Ключові слова: генерація тексту в зображення, глибинні нейронні мережі, дифузійні моделі, донавчання

Petrov Mykola. The task of text-to-image generation using neural networks.

This bachelor's thesis examines the task of text-to-image generation using modern deep neural networks. The work provides an overview of key generative approaches, including GANs, VAEs, and diffusion models, with particular emphasis on the architecture of Stable Diffusion – a latent diffusion model that combines components such as CLIP, U-Net, and a VAE-based decoder. Special

attention is given to fine-tuning methods, such as Textual Inversion and DreamBooth, which allow for adapting pretrained models to generate new objects based on a small number of training examples. In the practical part of the thesis, the Stable Diffusion v1.5 model was fine-tuned on a custom dataset containing 15 images of a target object – a plush toy – along with corresponding textual descriptions. The experimental results demonstrate the effectiveness of personalized generation: the model was able to reproduce the target object in various styles, scenes, and contexts while maintaining its characteristic visual features, highlighting the potential of this approach for further use in creative and applied computer vision tasks.

Keywords: text-to-image generation, deep neural networks, diffusion models, fine-tuning

Зміст

1. Вступ

Генерація зображень на основі текстового опису – це один із найяскравіших прикладів того, як сучасні нейронні мережі здатні моделювати складні зв’язки між різнорідними типами даних. За останнє десятиліття для вирішення цієї задачі з’явилося багато моделей з відкритим кодом, які не вимагають спеціальних дозволів на використання та сумісні зі звичайним користувацьким обладнанням.

Такий прорив став можливим завдяки зростанню обчислювальної потужності, появі великих наборів даних високоякісних зображень із точними текстовими описами, а також розвитку математичної теорії – зокрема, удосконаленню архітектур нейронних мереж, оптимізаційних алгоритмів і моделей узгодження між текстом і зображенням. Наприклад, однією з важливих особливостей навчання моделей генерації тексту в зображення є поєднання тексту і зображення у спільному латентному (прихованому) просторі, де враховується контекст, значення слів і навіть стиль. Це дає змогу подати обидва типи даних у єдиному компактному представленні меншої розмірності, що, у свою чергу, полегшує подальше навчання моделі та зменшує обчислювальні витрати.

Загалом застосування подібних систем присутнє в багатьох галузях – від креативних індустрій до наукових візуалізацій. До прикладу, у сфері дизайну їх використовують для швидкого створення ескізів, варіантів оформлення інтерфейсів або навіть цілих сцен. У сфері комп’ютерних ігор – для генерації унікальних персонажів, оточення чи текстур. Художники та ілюстратори застосовують ці системи як інструмент натхнення або як допоміжний засіб у побудові візуального наративу.

Типовою особливістю систем генерації зображень на основі текстового опису є здатність моделі запам’ятовувати загальні класи об’єктів і стилів, що були представлені в тренувальному датасеті. Для цього дані мають бути достатньо різноманітними, а процес навчання – тривалим. Якщо сформований датасет містить лише зображення конкретного об’єкта або

стилю, то за допомогою донавчання модель можна навчити розпізнавати та відтворювати саме ці приклади. У подальшому вона буде здатна генерувати їх у різних контекстах: у стилі живопису, в новому середовищі, з іншою емоцією чи дією. Таке донавчання відкриває новий підхід до створення віртуальних персонажів (аватарів), анімацій та AR/VR-додатків, де важливо зберігати візуальну ідентичність за високої варіативності сцен.

Метою цієї роботи стало дослідження та реалізація підходу до генерації тексту в зображення з урахуванням адаптації моделі до конкретного об’єкта на основі невеликої вибірки його фотографій.

Упродовж роботи було проаналізовано сучасні підходи до генерації зображень; розібрано архітектуру дифузійних моделей з доданим текстовим описом, зокрема Stable Diffusion; досліджено методи перенавчання великих моделей на малих вибірках; здійснено донавчання моделі на обраному датасеті із зображеннями конкретного об'єкта; оцінено здатність моделі до генерації нового персоналізованого контенту.

2. Архітектура сучасних моделей генерації зображень

2.1. Generative Adversarial Networks (GAN). Одним із перших успішних підходів до генерації фотореалістичних зображень стали GAN, запропоновані у 2014 році Ієном Гудфеллоу в роботі [1]. Вони складаються з двох нейронних мереж – генератора і дискримінатора, які навчаються у змагальному процесі:

- Генератор (G) прагне створювати зображення, подібні до справжніх, тобто наданих у датасеті;
- Дискримінатор (D) – відрізняти справжні зображення від тих, що були згенеровані генератором.

Процес навчання GAN формалізується як гра з нульовою сумою, де генератор мінімізує функцію втрат [1], яку дискримінатор намагається максимізувати:

$$\min_G \max_D \left\{ E_{x \sim p_{data}} [\log D\,(x)] + E_{z \sim p_z} \left[\log \left(1 - D\big(G(z)\big) \right) \right] \right\},$$

де:

- $x \sim p_{\text{data}}$ – зображення з датасету;
- $z \sim p_z$ – випадковий вектор шуму, зазвичай стандартного нормального розподілу;
- $G(z)$ – згенероване генератором зображення;
- $D(x)$ – оцінка ймовірності того, що зображення з датасету є справжнім;
- $D\big(G(z)\big)$ – оцінка ймовірності того, що згенероване зображення генератором є справжнім.

Така конструкція дозволяє моделі навчатися генерувати доволі складні розподіли зображень, проте варто звернути увагу і на особливості інтеграції текстових описів.

У більшості випадків текст подається у вигляді фіксованого векторного представлення, отриманого за допомогою кодувальника, такого як Recurrent Neural Network (RNN) або трансформера. Цей вектор

використовується як умова для генератора та/або дискримінатора. Деякі моделі, як-от AttnGAN [2], впроваджують механізми уваги для кращого узгодження між словами тексту та ділянками зображення. Однак навіть із такими вдосконаленнями GAN стикаються з низкою обмежень під час генерації зображень на основі тексту.

Наприклад, генератор може втрачати здатність до варіативної генерації через порушення балансу з дискримінатором, що призводить до повторюваних результатів і обмеженої різноманітності зображень. Навіть з використанням механізму уваги, GAN мають обмежені можливості локального узгодження окремих слів опису з відповідними візуальними елементами, що ускладнює точне відтворення змісту тексту. Через те, що GAN не формують явного розподілу ймовірностей для вихідних даних, ускладнюється контроль за процесом генерації та стає важко впроваджувати такі моделі в більш складні ймовірнісні системи.

2.2. Variational Autoencoders (VAE). Ще одним поширеним підходом до генерації зображень є VAE, запропоновані у 2013 році Дідеріком Кінгмою та Максом Велінгом у роботі [3]. На відміну від GAN, де генерація здійснюється через змагання двох мереж, VAE навчається відновлювати зображення з латентного простору та моделює ймовірнісний розподіл ознак, які лежать в основі цих зображень. Архітектура VAE складається з двох основних компонентів:

- Кодувальник $q_\phi(z|x)$: перетворює зображення x у латентне представлення z,
- Декодувальник $p_\theta(x|z)$: відновлює зображення з латентного вектора z.

Навчання VAE відбувається шляхом максимізації нижньої оцінки правдоподібності (Evidence Lower Bound, ELBO) [3], що поєднує точність реконструкції та регуляризацію латентного простору:

$$log\, p(x) \geq \mathcal{L}_{VAE}(\theta, \phi; x) = E_{z\sim q_\phi(z|x)}[\log p_\theta\,(x|z)] - D_{KL}\left(q_\phi(z|x)\,\middle|\,p(z)\right),$$

де:

- x – зображення з датасету;
- z – латентний вектор (приховане представлення);
- $q_\phi(z|x)$ – апроксимація апостеріорного розподілу $\mathrm{p}(z|x)$, що моделюється кодувальником;
- $p_\theta(x \mid z)$ – реконструкція зображення через декодувальник;
- $p(z)$ – апріорний розподіл у латентному просторі;
- $E_{z\sim q_\phi(z|x)}[\log p_\theta\,(x|z)]$ – якість реконструкції зображення;
- $D_{KL}\left(q_\phi(z|x) \mid p(z)\right)$ – дивергенція Кульбака–Лейблера, яка змушує згенеровані латентні змінні бути схожими на $p(z)$.

Окремим розвитком цієї архітектури є так звані рекурентні VAE, як-от DRAW [4], у яких процес генерації зображення відбувається поступово – шляхом послідовного оновлення внутрішнього полотна протягом кількох ітерацій. На кожному кроці така модель використовує нову латентну змінну, що дозволяє уточнювати локальні деталі зображення, моделювати довготривалі залежності та поетапно нарощувати складність сцени. Такий підхід відрізняється від класичних VAE тим, що ніби наближає процес генерації до послідовного «малювання» зображення.

Модель alignDRAW [5] розширює підхід DRAW та додає механізм уваги до текстового опису, що дає змогу враховувати змістову структуру запиту на кожному етапі побудови зображення. На вході текст подається не як єдиний вектор, а як послідовність векторних представлень, отриманих з двоспрямованої RNN. У процесі генерації модель на кожному кроці визначає, які слова текстового опису є найбільш релевантними на даному етапі, і враховує їх з різною інтенсивністю. Завдяки цьому формується контекстуальне представлення, що дозволяє узгодити вміст зображення з відповідними частинами опису.

Попри здатність до локалізованого узгодження між словами опису й окремими частинами зображення, ці моделі мають характерне для VAE обмеження – розмитість зображень, зумовлену ймовірнісною природою вибірки з латентного простору. Для покращення якості генерацій alignDRAW застосовує окрему постобробку на основі Laplacian pyramid GAN (LAPGAN) [6], яка підвищує чіткість, але не є частиною процесу навчання. Це обмежує ефективність моделі в реальних сценаріях застосування.

2.3. Diffusion Models. Дифузійні моделі, запропоновані у 2020 році Джонатаном Хо, Аджаєм Джайном та Пітером Аббілем у роботі [7], стали справжнім проривом у галузі генерації зображень завдяки здатності створювати реалістичні, деталізовані та структурно узгоджені зображення за високої стабільності навчання. Ідея архітектури моделі полягає у таких двох фазах:

- Прямий процес: поступове додавання випадкового шуму до зображення x_0, поки воно не перетвориться на чистий шум x_T. Цей процес задається як ланцюг Маркова [7]:

$$q(x_t \mid x_{t-1}) = \mathcal{N}\left(x_t;\ \sqrt{1-\beta_t}\, x_{t-1},\ \beta_t I\right),$$

де:

  - x_t – зашумлене зображення на кроці t;
  - β_t – дисперсія шуму на кроці t.

- Зворотний процес: поступове відновлення зображення x_0 із повністю зашумленого стану x_T шляхом поетапного усунення шуму. Цей процес апроксимується параметризованою нейронною мережею та задається як ланцюг Маркова [7]:

$$p_\theta(x_{t-1} \mid x_t) = \mathcal{N}\left(x_{t-1};\ \mu_\theta(x_t, t),\ \Sigma_\theta(x_t, t)\right),$$

де:

- $\mu_\theta(x_t, t)$ та $\Sigma_\theta(x_t, t)$ – параметри відновленого розподілу на кроці t.

Функцію витрат відповідно пропонується визначати як середньоквадратичну похибку (mean squared error, MSE) між справжнім шумом, доданим до зображення, та його оцінкою, згенерованою моделлю [7]:

$$\mathcal{L}_{simple}(\theta) = \mathrm{E}_{t,\mathrm{x}_0,\varepsilon}\left[\left|\varepsilon - \varepsilon_\theta\left(\sqrt{\bar{\alpha}_\mathrm{t}}\mathrm{x}_0 + \sqrt{1-\bar{\alpha}_\mathrm{t}}\varepsilon, \mathrm{t}\right)\right|^2\right],$$

де:

- $\varepsilon \sim \mathcal{N}(0, I)$ – випадковий шум,
- $\varepsilon_\theta(\cdot, t)$ – передбачений моделлю шум,
- $\alpha_t \coloneqq 1 - \beta_t \ \wedge \ \overline{\alpha_t} \coloneqq \prod_{s=1}^{t} \alpha_s$.

Дифузійні моделі добре масштабуються та дозволяють гнучко контролювати процес генерації. На відміну від VAE, вони забезпечують чіткіші й менш згладжені результати, а порівняно з GAN – менш схильні до проблем нестабільної оптимізації та втрати різноманіття згенерованих зображень. У подальшому розділі буде розглянуто модифікацію дифузійної моделі, яка дозволяє здійснювати керовану генерацію за текстовим описом на прикладі моделі Stable Diffusion.

3. Stable Diffusion

Після загального огляду принципів дифузійного підходу варто зосередитися на його практичному втіленні в задачах генерації зображень на основі текстового опису. Одним із таких рішень є модель Stable Diffusion, запропонована Робіном Ромбахом та колегами у 2022 році в роботі [8]. Це так звана латентна дифузійна модель (Latent Diffusion Model, LDM), в якій генерація відбувається не в піксельному просторі, а в стислому латентному представленні, що дає змогу значно знизити обчислювальні витрати. Латентний простір визначається тут як багатовимірний простір прихованих ознак, у якому дані представлені в узагальненому та компактному вигляді, що зберігає ключову семантичну інформацію. Модель має модульну структуру й складається з трьох основних компонентів: текстового кодувальника, дифузійного модуля (U-Net) та декодувальника, який реконструює зображення з латентного простору.

3.1. Текстовий кодувальник. На вході текстовий промпт спочатку розбивається на окремі токени за допомогою токенізатора, побудованого на основі алгоритму Byte Pair Encoding (BPE).

Під час попереднього навчання токенізатора на великому текстовому корпусі відбувається поступове об’єднання найвживаніших пар символів у токени більшої довжини, що дозволяє ефективно сегментувати текст на зручні для обробки фрагменти. При цьому такий підхід забезпечує здатність моделі працювати навіть з рідкісними або новими словами.

Після токенізації кожен токен кодується у вектор чисел (word embedding), який зберігає інформацію про його зміст і використовується моделлю для подальшої обробки, аналогічно до підходу, що застосовується в GPT-2 [11]. Отримана послідовність векторів обробляється трансформером, який виступає текстовим кодувальником моделі Contrastive Language–Image Pretraining (CLIP), розробленої Алеком Редфордом та співавторами у 2021 році в роботі [9].

Загалом CLIP складається з двох окремих кодувальників: візуального (наприклад, ResNet-50 або Vision Transformer) та текстового на базі трансформера. Під час навчання CLIP була оптимізована таким чином, щоб латентні вектори зображень і відповідних їм текстових описів мали максимально близьке представлення у спільному латентному просторі. Це досягається шляхом максимізації скалярного добутку (cosine similarity) між відповідними парами зображення–текст.

Для отримання єдиного векторного представлення всього тексту використовується вихід трансформера, що відповідає спеціальному токену [EOS], який позначає кінець послідовності [9]. Саме цей вектор і використовується як умовний сигнал (conditioning vector) для дифузійної моделі, впливаючи на процес генерації відповідного зображення.

3.2. U-Net. Центральним елементом дифузійної моделі є нейронна мережа, яка здійснює зворотний процес очищення зображення від шуму. У Stable Diffusion для цієї мети використовується модифікований варіант моделі U-Net, який виступає як основний генератор.

Базова версія U-Net була запропонована Олафом Роннеберґом, Філіппом Фішером та Томасом Броксом у 2015 році в роботі [10]. Її архітектура показана нижче на Рис.1, який було представлено в роботі [10].

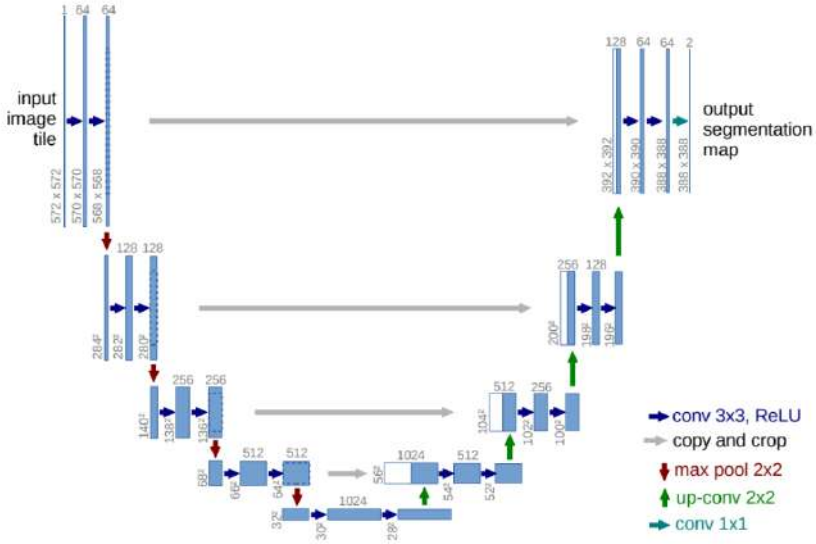

Рис. 1. Архітектура U-Net (приклад для зображення розміром 32×32 пікселі на найнижчому рівні просторової роздільності). Кожен синій блок представляє багатоканальний тензор ознак. Кількість каналів зазначено над блоком. Розміри по осях x та y вказано в нижньому лівому куті кожного блоку. Білі блоки відповідають скопійованим тензорам ознак. Стрілки позначають різні типи операцій. [10].

Першим компонентом U-Net є так званий шлях зменшення просторової роздільності (downsampling path) – послідовність згорткових блоків, які зменшують просторову роздільність вхідного зашумленого зображення, одночасно збільшуючи кількість каналів. Це дає змогу виділяти ознаки об’єкта на різних масштабах.

У центральній частині мережі розташований вузький блок (bottleneck), який працює з найкомпактнішим представленням зображення, акумулюючи та комбінуючи інформацію з усієї вхідної ділянки.

Далі йде етап відновлення просторової роздільності – так званий шлях розгортання (upsampling path), який є симетричним до попереднього етапу. На цьому етапі застосовуються транспоновані згортки або інші методи, що поступово збільшують розмірність зображення. Цей шлях поєднується з відповідними рівнями на етапі зменшення роздільності за допомогою пропускових з’єднань (skip connections), що дає змогу зберігати локальні деталі.

Після розгляду класичної архітектури U-Net, повернімося до її модифікованого варіанту, який використовується в моделі Stable Diffusion, як-от на Рис. 2, представленому в роботі [8].

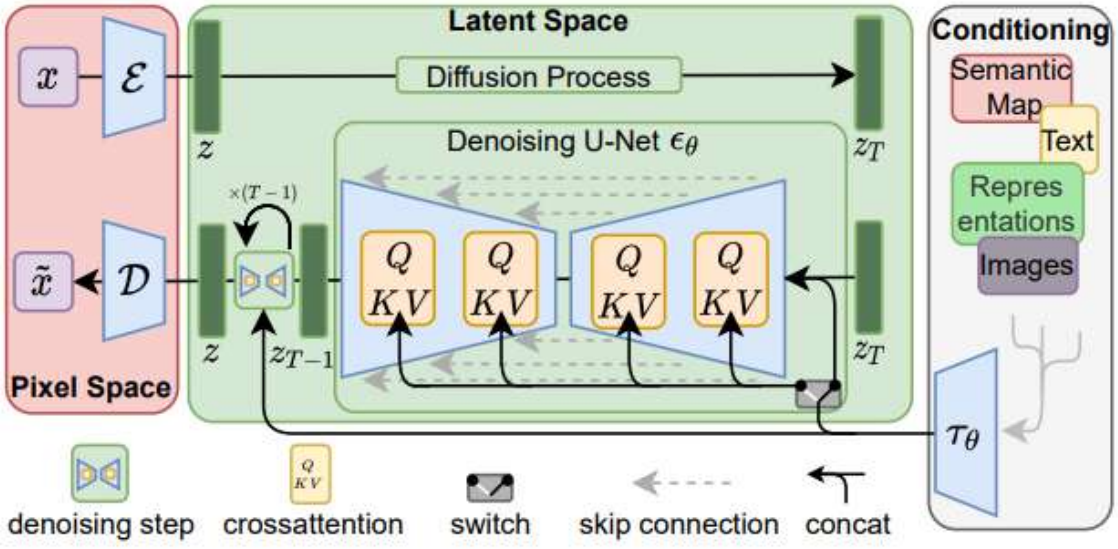

Рис. 2. На схемі зображена модель LDM, у якій додаткова інформація (наприклад, текст чи семантична карта) може враховуватися шляхом конкатенації або за допомогою механізму перехресної уваги. [8].

Окрім самого зображення, U-Net також отримує низку додаткових векторів, які впливають на процес генерації. Зокрема, це зашумлене латентне представлення зображення, семантичне векторне представлення текстового опису (отримане за допомогою текстового кодувальника), а також часовий вектор, що кодує поточний крок t у процесі дифузії. Усі ці сигнали впроваджуються на різних рівнях мережі, що дозволяє моделі точно узгоджувати згенероване зображення із заданим текстовим описом.

Модифікована архітектура U-Net також є зручною для додаткового навчання, оскільки має чітку модульну структуру з великою кількістю окремих блоків, що дозволяє адаптувати лише окремі її частини без потреби повного перенавчання всієї моделі.

3.3. Декодування зображень. Оскільки Stable Diffusion є латентною дифузійною моделлю, основні обчислення під час генерації відбуваються не

в піксельному, а в компактнішому латентному просторі. Це дозволяє суттєво знизити обчислювальні витрати: замість обробки повнорозмірних тензорів (наприклад, 512×512×3), модель працює з набагато меншими латентними представленнями, зазвичай розміром 64×64×4.

Проте для отримання фінального зображення необхідно здійснити зворотне перетворення з латентного простору до простору RGB-зображення. Для цього використовується декодувальна нейронна мережа – спеціальний модуль, який виконує реконструкцію. У моделі Stable Diffusion цю функцію виконує декодувальник VAE. Він забезпечує якісне відновлення зображення, зберігаючи основні візуальні характеристики, закладені в латентному представленні. Формально, процес відновлення фінального зображення $\hat{x}$ можна описати як:

$$\hat{x} = D(z),$$

де:

- $z \sim UNet(x_t, t, c)$ – латентне представлення, згенероване дифузійним процесом;
- c – умовний вектор з текстового кодувальника;
- D – декодувальник VAE.

Таким чином, повний цикл генерації в моделі Stable Diffusion включає обробку текстового запиту, генерацію в латентному просторі та реконструкцію зображення. У наступному розділі буде розглянуто, як використовувати цей процес у донавчанні.

4. Fine-tuning

4.1. Навіщо потрібне донавчання? Генеративні моделі сьогодні є системами, натренованими на великих об’ємах даних, здебільшого з відкритих інтернет-джерел. Тренувальні набори даних становлять від сотень мільйонів до мільярда прикладів, як-от LAION-5B dataset [15]. Для проведення базового навчання на такому датасеті потрібно мати величезні обчислювальні ресурси, і таке тренування може тривати десятки або навіть сотні годин на кластерах з десятків графічних процесорів. Наприклад, базове тренування моделі Stable Diffusion v1 потребувало близько 150 000 GPU-годин на графічних процесорах NVIDIA A100 [13], що еквівалентно безперервній роботі 100 таких GPU протягом понад двох місяців.

У донавчанні моделі за новим спеціалізованим завданням неефективно запускати процес навчання з нуля на оновленому датасеті. Натомість застосовується transfer learning – стратегія машинного навчання, за якої знання, отримані під час вирішення однієї задачі, повторно використовуються для вирішення іншої, пов’язаної задачі. Це дозволяє уникнути повторного навчання всієї моделі, що значно зменшує витрати часу та ресурсів.

Fine-tuning є конкретним різновидом transfer learning, коли попередньо натренована модель адаптується до нової специфічної задачі шляхом додаткового навчання на меншому, спеціалізованому датасеті. У цьому процесі модель частково або повністю оновлює свої параметри, зберігаючи набуті раніше узагальнені знання.

Такий підхід пропонується застосовувати для задач персоналізованої генерації, зокрема, відтворення конкретних об’єктів за обмеженим набором зразків. У наступному розділі буде розглянуто приклад такої адаптації.

4.2. Адаптації до нових даних. У випадках персоналізованої генерації зображень, коли модель має навчитися відтворювати конкретний об’єкт, наприклад, предмет, персонаж або обличчя, важливим є не лише

досягнення зовнішньої схожості, але й забезпечення сталого образу в різноманітних контекстах. Модель повинна зберігати характерні риси нового об’єкта незалежно від умов сцени, стилю чи освітлення. Реалізація цього завдання ускладнюється кількома важливими чинниками.

Насамперед, ідеться про обмежений обсяг даних, на яких виконується донавчання. Іноді для адаптації до нового об’єкта доступні лише кілька прикладів. За таких умов модель схильна до надмірного пристосування (overfitting), тобто до простого запам’ятовування навчальних прикладів без здатності узагальнювати на нові ситуації.

Крім того, під час повного донавчання часто спостерігається ефект стирання знань (catastrophic forgetting), коли модель втрачає попередні знання, набуті під час базового навчання. Це може виражатися у зниженні якості генерацій для інших об’єктів та стилів, з якими раніше модель працювала впевнено.

Ще однією проблемою є змішування стилів: навіть коли модель зберігає форму об’єкта, вона може неусвідомлено переносити на нього стилістичні риси з інших частин початкового навчання, змінюючи кольори, пропорції або текстури на більш «звичні» для себе.

Також після донавчання модель не завжди стабільно реагує на текстові запити. Навіть коли використовується спеціальний тег для нового об’єкта, результати можуть бути нестійкими або непередбачуваними. Це може свідчити про те, що асоціація між текстовим описом і візуальним образом об’єкта залишається недостатньо закріпленою.

У наступних підрозділах буде розглянуто два популярні методи донавчання – Textual Inversion та DreamBooth. Обидва підходи спрямовані на покращення персоналізованої генерації та підтримку загальних можливостей моделі.

4.3. Textual Inversion. Метод Textual Inversion дозволяє моделі асоціювати новий текстовий маркер із певним візуальним образом без зміни

основних параметрів моделі. Реалізація підходу передбачає введення до словника моделі нового токена, для якого ініціалізується окремий вектор у просторі текстових векторних представлень (word embedding space). Під час навчання оновлюється лише цей вектор, тоді як усі інші параметри моделі залишаються незмінними. Щоб підвищити асоціативність із конкретною категорією об’єктів, поруч із новим токеном у текстових описах часто додають назви споріднених класів, до яких належить сам об’єкт.

При цьому якість генерації суттєво залежить від сформульованих текстових промптів. З огляду на це об’єкт може змішуватися з іншими стилями, притаманними базовій моделі, оскільки донавчання виконується без змін у її генеративній частині.

4.4. DreamBooth. Метод DreamBooth, запропонований Норбертом Руїсом та співавторами у 2022 році в роботі [12], дає змогу адаптувати дифузійні моделі генерації зображень до конкретного нового об’єкта на основі лише кількох прикладів. На відміну від Textual Inversion, цей підхід передбачає також оновлення параметрів всієї генеративної складової моделі (наприклад, U-Net) або її частини, що дозволяє точніше передати візуальні особливості об’єкта та сформувати стійкий зв’язок між текстом і зображенням.

Під час тренування модель отримує текстові описи, в які вбудовується унікальний маркер разом із загальною категорією об’єкта. Щоб запобігти втраті попередніх знань, навчання здійснюється як на цільових зображеннях, так і на зразках із базового класу, до якого належить об’єкт. Це зменшує ризик стирання знань і допомагає зберегти здатність моделі до генерації інших прикладів того ж класу. Додатково використовуються спеціальні регуляризаційні функції втрат, що сприяють збереженню властивостей класу.

Метод також передбачає розширення навчального набору за допомогою аугментацій – змін ракурсу, кольору, освітлення та ін., що

покращує здатність моделі узагальнювати. У результаті DreamBooth забезпечує високу точність передачі образу, стабільність у різних умовах і ефективність навіть за мінімальної кількості навчальних даних.

У наступному розділі буде представлено застосування методу DreamBooth на практиці для адаптації моделі Stable Diffusion до нового об’єкта за допомогою невеликої вибірки прикладів.

5. Практична частина

Практична частина була реалізована на мові Python через Jupyter Notebook в середовищі Google Colaboratory. Ця платформа дозволяє писати програмний код через веб-інтерфейс та компілювати його на віддаленому сервері. Об’єм серверної RAM: 53.0 GB. Параметри серверного графічного процесора (GPU):

- CUDA Version: 12.4;
- GPU name: NVIDIA L4;
- GPU memory: 22.5 GB.

5.1. Підготовка датасету. У межах цієї роботи було зібрано власний датасет, що включає 15 фотографій 512×512 цільового об’єкта, зроблених з різних ракурсів, у різних умовах освітлення та на різному фоні. Для кожного зображення було підготовлено текстовий опис, що відповідає його змісту.

У ролі цільового об’єкта виступає м’яка іграшка, що в текстових описах позначається як mysks plush toy. Тут mysks є спеціальним токеном, який модель має навчитися асоціювати з відповідним візуальним образом. Цей токен був обраний таким чином, щоб він не зустрічався або майже не зустрічався у базовому тренувальному корпусі моделі, і не мав попередніх асоціацій.

Натомість plush toy виконує функцію класу об’єкта, що спрощує задачу узагальнення, оскільки він належить до семантичних категорій, на яких модель вже проходила попереднє базове тренування. Декілька прикладів зображень з датасету із відповідними текстовими описами наведено нижче (Рис. 3-6).

Рис.3. a photo of mysks plush toy walking on stacked wooden planks

Рис.4. a photo of mysks plush toy in front of glowing tents inside a building

Рис.5. a photo of mysks plush toy standing on a wooden post near a silver car

Рис.6. a photo of mysks plush toy on top of a wrapped pallet near a modern building

Після підготовки й завантаження датасету на сервер наступним кроком стало завантаження та донавчання моделі.

5.2. Донавчання моделі. Базовою моделлю було обрано Stable Diffusion v1.5, а основним методом донавчання – DreamBooth. Було використано офіційний скрипт train_dreambooth.py [14] з бібліотеки diffusers (Hugging Face), до якого були внесені незначні зміни для більш зручної обробки підготовленого датасету. Упродовж етапу донавчання, для

точнішого врахування тегу mysks, відбувалося також навчання його векторного представлення. Тренувальні гіперпараметри:

- Batch size: 8
- Gradient accumulation: 1
- Learning rate: 2e-6
- Mixed precision: bf16
- Загальна кількість кроків: 1600
- Тип оптимізатора: 8-бітний AdamW

Такий набір параметрів було визначено емпірично, виходячи з якості генерації та стабільності навчального процесу.

5.3. Результати. Під час базового навчання модель не стикалася з зображеннями нашої м’якої іграшки і не мала змоги побудувати асоціації ні з її точним візуальним образом, ні з відповідним спеціальним тегом. Як наслідок, генерація зображень за такими промптами є для нас нерелевантною. Це проілюстровано на зображеннях, згенерованих базовою моделлю Stable Diffusion v1.5 (Рис. 7, 8).

Рис.7. a photo of mysks walking on a snowy mountain trail at sunset

Рис.8. a cinematic portrait of a plush toy of a wild boar piglet during golden hour near the lake

Процес донавчання моделі тривав орієнтовно 45–60 хвилин. Після нього вона навчилася відтворювати характерні риси об’єкта – його форму, колір і текстуру – у відповідь на текстові промпти з тегами mysks та mysks plush toy. Нижче наведено приклади згенерованих зображень моделлю після донавчання разом із відповідними текстовими описами (Рис. 9-16).

Рис.9. a cinematic portrait of mysks plush toy during golden hour near the lake

Рис.10. a photo of mysks plush toy wearing a tiny astronaut suit inside a spaceship

Рис.11. a photo of mysks near books

Рис.12. a photo of mysks walking on a snowy mountain trail at sunset

Рис.13. an under water photo of mysks plush toy in aquarium

Рис.14. a reaslistic photo of mysks sitting on a wooden floor in a Scandinavian-style living room, shallow depth of field

Рис.15. a 3D render of mysks, studio lighting

Рис.16. a watercolor painting of mysks in a flower field

На зображеннях також видно, що дотренована модель здатна змінювати вигляд іграшки залежно від опису, зберігаючи при цьому її характерні риси. Образ іграшки з’являється в різних умовах – на вулиці, в кімнаті, в акваріумі, у снігу чи в космічному кораблі. Також змінюються стиль зображення (наприклад, фото, 3D-рендер чи акварель), освітлення (захід сонця, студійне світло) та поза об’єкта.

Проведене дослідження показало, що генерація тексту в зображення після етапу fine-tuning має достатню стабільність. У більшості випадків модель коректно реагує на спеціальний тег і передає візуальні ознаки об’єкта.

Водночас, слід відзначити, що іноді спостерігаються незначні спрощення або втрати деталей у складніших запитах, зокрема у разі взаємодії кількох факторів, як-от поза, стиль і фон. Крім того, модель у деяких випадках частково змішує вигляд іграшки з образом тваринки, яку ця іграшка уособлює. Як видається, ці недоліки можливо усунути за рахунок ще тривалішого донавчання та розширення навчального набору прикладів.

6. Висновки

У процесі виконання цієї кваліфікаційної роботи було розглянуто сучасні підходи до генерації зображень за текстовим описом із подальшою адаптацією моделі до нових даних. Основну увагу було приділено аналізу архітектури дифузійних моделей та можливостям їх донавчання за обмеженої кількості навчальних прикладів.

Практична частина дослідження продемонструвала, що навіть за незначного обсягу даних модель Stable Diffusion можна ефективно адаптувати до нового об’єкта за допомогою методу донавчання DreamBooth. Модель навчилася асоціювати спеціальний текстовий маркер із візуальним образом м’якої іграшки і відтворювати його в нових контекстах, зберігаючи характерні риси.

Попри досягнуті результати, у роботі також було виявлено низку обмежень, зокрема залежність якості генерації від точності текстових описів, а також тенденцію до спрощення деталей або стилістичних змін у складних візуальних ситуаціях на зображенні. Ці аспекти можуть бути предметом для подальших досліджень. Також перспективним напрямом є порівняння альтернативних методів донавчання, автоматизована оптимізація текстових промптів, впровадження додаткових механізмів контролю за стилем і композицією зображення.

Отже, результати роботи підтверджують доцільність використання методів донавчання для генерації персоналізованого візуального контенту й відкривають можливості для подальших експериментів у цьому напрямі.

Список використаних джерел

1. Goodfellow I. J., Pouget-Abadie J., Mirza M., Xu B., Warde-Farley D., Ozair S., Courville A., Bengio Y. Generative Adversarial Networks // *arXiv:1406.2661v1 [stat.ML]*. 10 Jun 2014. https://arxiv.org/pdf/1406.2661
2. Xu T., Zhang P., Huang Q., Zhang H., Gan Z., Huang X., He X. AttnGAN: Fine-Grained Text to Image Generationwith Attentional Generative Adversarial Networks // *arXiv:1711.10485v1 [cs.CV]*. 28 Nov 2017. https://arxiv.org/pdf/1711.10485
3. Kingma D.P., Welling M. Auto-Encoding Variational Bayes // *arXiv:1312.6114v11 [stat.ML] 10 Dec 2022.* https://arxiv.org/pdf/1312.6114
4. Gregor K., Danihelka I., Graves A., Rezende D. J., Wierstra D. DRAW: A Recurrent Neural Network For Image Generation // *arXiv:1502.04623v2 [cs.CV.]* 20 May 2015. https://arxiv.org/pdf/1502.04623
5. Mansimov E., Parisotto E., Ba J. L., Salakhutdinov R. Generating Images from Captions with Attention // *arXiv:1511.02793v2 [cs.LG]*. 29 Feb 2016. https://arxiv.org/pdf/1511.02793
6. Denton E., Chintala S., Szlam A., Fergus R. Deep Generative Image Models using a Laplacian Pyramid of Adversarial Networks // *arXiv:1511.02793v2 [cs.LG]*. 29 Feb 2016. https://arxiv.org/pdf/1506.05751
7. Ho J., Jain A., Abbeel P. Denoising Diffusion Probabilistic Models // *arXiv:2006.11239v2 [cs.LG]*. 16 Dec 2020. https://arxiv.org/pdf/2006.11239
8. Rombach R., Blattmann A., Lorenz D., Esser P., Ommer B. High-Resolution Image Synthesis with Latent Diffusion Models // *arXiv:2112.10752v2 [cs.CV]*. 13 Apr 2022. https://arxiv.org/pdf/2112.10752
9. Radford A., Kim J. W., Hallacy C., Ramesh A., Goh G., Agarwal S., Sastry G., Askell A., Mishkin P., Clark J., Krueger G., Sutskever I. Learning Transferable Visual Models From Natural Language Supervision.// *arXiv:2103.00020v1 [cs.CV]*. 26 Feb 2021. https://arxiv.org/pdf/2103.00020

10. Ronneberger O., Fischer P., Brox T. U-Net: Convolutional Networks for Biomedical Image Segmentation // *arXiv:1505.04597v1 [cs.CV]*. 18 May 2015. https://arxiv.org/pdf/1505.04597

11. Radford A., Wu J., Child R., Luan D., Amodei D., Sutskever I. Language Models are Unsupervised Multitask Learners // *OpenAI*. 2019. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

12. Ruiz N., Li Y., Jampani V., Pritch Y., Rubinstein M., Aberman K. DreamBooth: Fine Tuning Text-to-Image Diffusion Models for Subject-Driven Generation // *arXiv:2208.12242v2 [cs.CV]*. 15 Mar 2023. https://arxiv.org/pdf/2208.12242

13. You Y. Diffusion Pretraining and Hardware Fine-Tuning Can Be Almost 7X Cheaper! Colossal-AI's Open Source Solution Accelerates AIGC at a Low Cost. 2022. URL: https://medium.com/%40yangyou_berkeley/diffusion-pretraining-and-hardware-fine-tuning-can-be-almost-7x-cheaper-85e970fe207b (дата звернення: 20.04.2025).

14. diffusers/examples/dreambooth/train_dreambooth_lora_sdxl.py - URL: https://github.com/huggingface/diffusers/blob/main/examples/dreambooth/train_dreambooth.py (дата звернення: 20.04.2025).

15. Beaumont R. LAION-5B: a new era of open large-scale multimodal datasets. 2022. URL: https://laion.ai/blog/laion-5b/ (дата звернення: 20.04.2025).