画像生成AIモデル

oboro:

技術報告書

AiHUB株式会社

AiHUB GENIAC

はじめに

本プロジェクトの目的と概要

本プロジェクトは、経済産業省（METI）および国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）の事業である「ポスト5G情報通信システム基盤強化研究開発事業／競争力ある生成AI基盤モデルの開発（GENIAC）」の第2期採択事業として実施されました。日本のアニメ製作現場で問題になっている人手不足などの課題を解決するため、フルスクラッチで画像生成モデルの開発を目指すものです。本書は開発した画像生成モデル「**oboro:**」の技術詳細を共有することを主目的としています。

開発した画像生成モデル「**oboro:**」の紹介

我々は新たな画像生成モデル「**oboro:**」を開発しました。著作権的にクリーンで安全な画像のみを学習データとすることを目的とし、フルスクラッチで構築されています。利用できるデータが限定的であることを考慮し、少ない画像数でも高品質な画像を生成できるよう設計されている点が大きな特徴です。基盤モデルの重み、推論コード、技術報告書を公開しています。

プロジェクトの意義

画像生成AIモデルに限らず、ほとんどのAIモデルは海外で開発が進んでいます。本プロジェクトはビジネス向けに開発された国産画像生成AIとして初めてオープンソースとしてモデルを公開しています。AiHUBは元々OSSコミュニティから生まれた企業です。開発過程を含めオープンにすることで、日本のAI研究者・技術者コミュニティに貢献し、国内のAI開発エコシステム全体の活性化を促進することを目指します。

モデル・推論コード

https://huggingface.co/aihub-geniac/oboro

技術報告書（本書）

AiHUB

目次

1. 開発背景

2022年8月のStable Diffusionの公開[1]は、テキストから画像を生成する画像生成AI (Text-to-Image AI) の歴史における大きな転換点となった。それまで一部の研究者や企業に限られていた高性能な画像生成技術が、オープンソースとして一般に開放されたことで、オープンソースソフトウェア(OSS)コミュニティによって開発が加速し、ビジネスの現場にも波及効果が現れ始めた。

初期のモデルでは明らかにAIが生成したかのような破綻した写真やイラストが出力されていたものの、驚異的なスピードで進化を繰り返している。画像生成モデルはすでに様々な場面で利用が始まっているものの、法整備や社会的なコンセンサスが得られないうちに使えるレベルに到達してしまったがゆえの問題も多く存在する。

1.1. 画像生成技術の向上

初期のモデルでは生成画像の品質も低く、テキストの指示にもあまり従わず、構図などを指定することも困難だった。近年のモデルでは組み合わせるテキストエンコーダを増やすことでテキストの理解が進んでいる。構図に関しては2023年に元画像の輪郭、ポーズ、深度情報などを保持したまま画像を生成するControlNet[2]が登場し、画像生成の自由度を飛躍的に向上させた。これにより、単なる「お絵描きAI」から、デザインや映像制作における「指示可能なツール」へと進化することとなった。

さらに、巨大なモデル全体を再学習することなく、特定のキャラクターや画風（絵柄）を追加学習させる効率的なファインチューニング手法LoRA (Low-Rank Adaptation)[3]が普及することで、個人が独自のモデルを容易に作成できるようになり、汎用性から特化モデルまで対応できるようになってきている。

2024年に入り、LLM（大規模言語モデル）で成功を収めたTransformerアーキテクチャを拡散モデルに統合するDiT (Diffusion Transformer) [4]のようなアプローチが主流になりつつある。Stable Diffusion 3[5,41]もこの流れを汲んでおり、従来のU-Netベースのモデルよりもテキスト解釈能力（プロンプトへの忠実性）が格段に向上し、より複雑で正確な画像生成が可能になっている。これら画像生成技術の詳細については2章で詳しく述べる。

1.2. 画像生成AIの倫理的課題とデータセットの制約

Stable Diffusionをはじめとする初期のモデルは、権利者の個別許諾取得を前提としない公開ウェブ由来の数億～数十億規模の画像・テキスト対を含む大規模データセットで学習されてきた。このような学習データの性質が、画像生成AIの開発をめぐる法的・倫理的な議論を喚起している。また、権利処理が確認できない素材や違法にアップロードされたコンテンツを含むおそれのあるデータセットで学習されたモデルについて、諸外国で権利者が訴訟を提起する例も生じており、法的リスクの存在が広く認識されつつある。各国の制度や運用は依然として統一されておらず、日本では著作権法第30条の4[50, 51]により、所定の要件を満たす限り権利者の個別許諾なしに学習に用いることも可能だが、条文解釈や著作権観の相違が残り、判例の蓄積も十分とはいえないため、実務上は、できるだけ権利面でのリスクが低いデータセットを選択するのが望ましい。画像生成AIの開発は、当初はOSSコミュニティを中心に研究用途や個人利用を念頭に進められてきたが、近時はビジネスの現場への導入も拡大している。その一方で、前記の論点に起因する摩擦が生じ、画像生成技術を活用した企業・公共団体が社会的批判や反発に直面する事例も見受けられる。

本研究では、アニメ制作の現場での利用を想定した画像生成モデルを開発するが、アニメーションの制作に、権利者の個別許諾を得ずに学習に用いた画像生成AIが使用される場合、その事実は視聴者から必ずしも肯定的に受け止められない可能性がある。また、クリエイターの立場からは、生成物が既存の著作物と偶然に類似するリスクが無視できないモデルの利用は難しいという課題もある。しかし、権利関係が明確で使いやすいと評価できる大規模データセットを整備することは容易ではない。このため、限られたデータでも学習可能なモデル・アーキテクチャの採用が必要となる。

1.3. アニメ製作現場へ導入する画像生成AIが持つべき性能と学習データセット

本プロジェクトで開発するモデルはアニメ製作会社が導入することを想定している。アニメ製作会社ごとに画像生成AIに求める性能が異なることや、機密情報の漏洩の観点からアニメ製作会社ごとにモデルを開発する必要がある。そこで基盤モデルとして著作権に配慮されたデータセットで学習されたモデルを開発し、アニメ製作会社ごとに基盤モデルに追加学習を行うという想定で開発を行った。これらを本プロジェクトでは「基盤モデル」「特化モデル」と呼ぶ。

本書において、著作権に配慮されたデータセットとはPublic domain、CC0、著作権者に学習の了解を得たものなどの画像で構成されたデータセットを言う。これらは画像ごとに確認する必要があるため、Stable Diffusionなどが利用したLAIONなどのデータセットに比べて画像数が少なくなる。画像が少ないと過学習になりやすかったり、生成品質が悪くなるなどの問題が生じるため、小さいデータセットに対応した新しいモデルの開発が必要になる。

2. 競合モデルの状況

2.1. 歴史的背景

初期の画像生成AI分野を牽引したのは、敵対的生成ネットワーク（Generative Adversarial Networks, GANs）[7]であった。GANは、画像を生成する「生成器（Generator）」と、その画像が本物か偽物かを見分ける「識別器（Discriminator）」を互いに競わせる敵対的学習の枠組みを用いる。このアーキテクチャは、一度学習が完了すれば非常に高速に画像を生成できるという利点を持ち、高忠実度な画像合成能力を示した。しかしその一方で、ナッシュ均衡解に到達するための学習プロセスが不安定であり、「モード崩壊（Mode Collapse）」と呼ばれる生成される画像の多様性が失われる問題や、勾配消失による学習の停滞といった課題を抱えていた。

これらの課題を克服する形で台頭したのが、拡散モデル（Diffusion Models）[8]である。拡散モデルは、物理学の非平衡熱力学に着想を得た生成手法であり、元画像に段階的にノイズを付加していく「前方過程（Forward Process）」と、その逆のプロセス、すなわち純粋なノイズから段階的にノイズを除去していく「後方過程（Reverse Process）」を学習することで画像を生成する。この反復的なデノイズ処理は、GANに比べて学習が非常に安定しており、生成されるサンプルの多様性や、複雑なデータ分布を正確に捉える能力において優位性を持つ。ただし、その反復的な性質上、推論（画像生成）に要する計算コストが高いというトレードオフが存在する。

現代の主要なテキスト画像生成モデル、例えばStable Diffusionシリーズ やMidjourney、DALL-E[9]などは、この拡散モデルのアーキテクチャを基盤として構築されている。これらのモデルは、単一の巨大なネットワークではなく、以下の3つの構成要素からなる。

- テキストエンコーダ（**Text Encoder**）: 入力されたテキストプロンプトを解釈し、モデルが理解可能な数値表現（埋め込み）に変換する。
- 画像推論アーキテクチャ（**Image Inference Architecture**）: テキストエンコーダからの情報（条件）を基に、潜在空間上でノイズ除去プロセスを実行し、画像を生成する画像生成モデルの主要部。
- 変分オートエンコーダ（**Variational Autoencoder, VAE**）: 高次元のピクセル空間と低次元の潜在空間との間で画像の圧縮・復元を行う。

2.2. テキストエンコーダ：意味理解の深化

人間が扱う言語・テキスト（あるいはプロンプト）は文字列で表されるが、ニューラルネットワークで取り扱うためには数値化（ベクトル化）する必要がある。テキストエンコーダは入力されたテキストを数値化する役割を持つ。テキストの内容・順序・関係性などを吟味して数値化を行えるかどうかが性能を左右する。生成画像がプロンプトに忠実に再現されるかどうかにとって重要な要素となる。テキストエンコーダにテキストを入力すると、条件付け(condictioning)と呼ばれるベクトルが出力される。

Stable Diffusion 1.5ではCLIP[10]が採用されたが、軽量がゆえに性能に限界があり、最新のモデルではCLIPに加えてT5, Gemmaなどのパラメータ数が多いモデルが採用されている。

2.2.1. CLIP：概念的アライメントの実現

CLIP (Contrastive Language-Image Pre-training) [10]は、OpenAIによって開発された、画像とテキストの関係性を学習するための画期的なモデルである。その最大の特徴は、4億もの画像とテキストのペアデータを用いた大規模な対照学習 (Contrastive Learning) にある。この学習プロセスでは、画像エンコーダとテキストエンコーダが、ペアとなる画像とテキストの埋め込みベクト

ル間のコサイン類似度が高くなるように、そしてペアでないもの同士の類似度が低くなるように、同時に訓練される。この結果、CLIPは画像とテキストを同じ意味を持つ多次元の「共有埋め込み空間」にマッピングする能力を獲得する。例えば、「犬の写真」というテキストの埋め込みは、実際の犬の画像の埋め込みと、この空間上で非常に近い位置に配置される。この特性により、CLIPは特定のタスクに対して再学習（ファインチューニング）を行うことなく、未知のタスクに対応する「ゼロショット学習」能力を発揮する。画像生成モデルにおいては、このCLIPの能力が、プロンプトに含まれる「概念」（例：「宇宙飛行士」、「馬」）や「スタイル」（例：「写真」、「アニメ風」）を、生成すべき画像の視覚的特徴と強力に結びつけるために利用される。Stable Diffusion 1.5やStable Diffusion XL（SDXL）[18]といった主要なモデルは、このCLIPをテキストエンコーダの中核として採用している。

2.2.2. T5：言語構造の精密な解釈

一方で、CLIPの複雑な構文の理解や、画像内への正確な文字描画（タイポグラフィ）には限界があった。この課題を解決するために注目されたのが、Googleによって開発されたT5 (Text-to-Text Transfer Transformer)[11]である。
T5は、翻訳、要約、質問応答など、あらゆる自然言語処理（NLP）タスクを「テキストからテキストへの変換問題」として統一的に扱うEncoder-Decoder型のTransformerアーキテクチャである。C4 (Colossal Clean Crawled Corpus) [12]と呼ばれる巨大なWebテキストコーパスで事前学習されており、言語の文法、構文、文脈といった構造的な側面を非常に深く理解する能力を持つ。画像生成モデルにおいてT5がもたらす最大の利点は、この高度な言語理解能力にある。例えば、「青い球体の上に乗った赤い立方体」といった複数のオブジェクト間の正確な空間関係や、「壁に『Stable Diffusion』と書かれたグラフィティ」といった具体的なテキストの描画指示を、T5はCLIPよりもはるかに正確に解釈できる。このため、Stable Diffusion 3やFLUX.1といった最新世代のモデルでは、T5がCLIPと併用されることが標準となっている。

2.2.3. Gemmaとその他のエンコーダ

近年の大規模言語モデル（LLM）のオープンソース化の流れは、テキストエンコーダの選択肢をさらに広げている。その代表例が、Googleによって開発された軽量なオープンモデルファミリーであるGemma[13]である。Gemmaは、テキストだけでなく、画像や音声といったマルチモーダルな入力を処理する能力を持ち、その高性能なテキスト理解能力から、新しい画像生成モデルのエンコーダとして採用され始めている。例えば、Lumina-Image 2.0[14]は、テキストエンコーダとしてGemma-2-2Bを採用しており、最先端のオープンLLMの能力を直接画像生成に活用するトレンドを象徴している。これらのテキストエンコーダの進化を俯瞰すると、一つの重要な方向性が見えてくる。それは、単一の万能なエンコーダに頼るのではなく、それぞれ異なる強みを持つ複数のエンコーダを組み合わせる「専門家委員会」のようなアプローチへの移行である。初期のStable Diffusion 1.5はCLIPのみを使用し、概念の表現には長けていたものの、複雑な構成や文字の描画には失敗しがちであった。SDXLは、サイズの異なる2つのCLIPモデルを組み合わせることで概念理解のニュアンスを深めたが、構造理解の根本的な問題は解決しなかった。真のブレークスルーは、Stable Diffusion 3やFLUX.1が採用した、CLIPとT5を組み合わせるハイブリッドアプローチによってもたらされた。この構成では、CLIPが「何を」描くべきか（概念、物体、美的スタイル）という視覚的な目標を設定し、T5 v1.1 XXLが「どのように」描くべきか（文法、構文、空間関係、スペリング）という構造的な指示を解読する。この役割分担により、各エンコーダの弱点を互いに補い合い、前例のないレベルでのプロンプト追従性とタイポグラフィ性能が実現されたのである。この傾向は、将来の最先端モデルが、単一の巨大なモデルではなく、専門化された複数のモデルの協調によって、さらなる高みを目指すことを示唆している。

2.3. 画像推論アーキテクチャ：U-NetからTransformerへ

テキストエンコーダによって解釈されたプロンプトの意図を、具体的な画像のピクセル情報へと変換するプロセスを担うのが、画像推論アーキテクチャである。この中核コンポーネントは、拡散モデルのデノイズ処理を実行するバックボーンであり、その設計はモデルの性能、スケーラビリティ、そして生成される画像の品質を根本的に決定づける。その進化は、畳み込みニューラルネットワーク（CNN）をベースとするU-Netから、より強力でスケーラブルなTransformerへの移行という、潮流を描いている。

2.3.1. U-Net：畳み込みネットワークの基礎

U-Net[15]は、元々医用画像セグメンテーションのために開発されたCNNベースのアーキテクチャである。その名称は、ネットワークの構造がU字型であることに由来する。エンコーダ（縮小パス）とデコーダ（拡大パス）から構成され、エンコーダで畳み込みとプーリングを繰り返して画像の特徴を抽出しながら空間的な解像度を下げ、デコーダで逆畳み込み（Transposed Convolution）によって解像度を復元していく。U-Netの独創的な点は、「スキップコネクション（Skip Connection）」と呼ばれる、エンコーダの各層の出力をデコーダの対応する層の入力に直接結合する仕組みにある。これにより、縮小パスで失われがちな高解像度の詳細な位置情報が、拡大パスに直接伝達され、精密な画像の再構成が可能となる。

拡散モデルの文脈では、このU-Netが各タイムステップにおいてノイズが付加された画像を入力として受け取り、付加されたノイズそのものを予測・除去する役割を担う。Stable Diffusion 1.5やSDXLといった、2023年頃までの主流な拡散モデルは、広域依存関係を直接捉えることができるself-attentionなどが組み込まれたU-Netの変種[8]を推論アーキテクチャの中核として採用していた。

2.3.2. DiT (Diffusion Transformer)：スケーラビリティの獲得

U-Netの帰納バイアスは拡散モデルにとって必須ではなく、自然言語処理や画像認識での豊富な知見が蓄積されたより単純な構造であるTransformerを拡散モデルのバックボーンに用いる研究[4]が成功を収め、以後の拡散モデルには当該研究にて提案されたDiT（Diffusion Transformer）とその変種が用いられることが多くなった。DiTはクラスラベルによる条件付けを採用する方式であり、実際の場面で要求されるテキスト条件付けに対応させる方法として、cross-attentionやjoint attentionなど、様々な機構が組み込まれる。

DiTでは、まずVAEによって圧縮された潜在空間上の画像を、さらに小さなパッチに分割し、それらをトークンのシーケンスとしてTransformerに入力する。これにより、Transformerは画像内の局所的な特徴（パッチ内）と大域的な文脈（パッチ間）の両方を効率的に学習することが可能になる。

DiTの最も重要な特性は、その優れた「スケーラビリティ」にある。モデルのパラメータ数や訓練に用いる計算量を増やすと、それに比例して生成画像の品質（FIDスコアで測定）が向上することが実験的に示されている。このスケーリング則は、より大規模なモデルを構築することで性能を継続的に向上させられることを意味しており、OpenAIのSoraやStable Diffusion 3といった次世代の超高性能モデルの基盤技術となっている。

2.4. VAE：潜在空間における表現と再構成

拡散モデル、特にStable Diffusionファミリーのような「潜在拡散モデル（Latent Diffusion Models, LDM）」において、VAE（Variational Autoencoder）は不可欠な役割を担っている。VAEは、モデルの計算効率と最終的な生成画質の両方に深く関与する、縁の下の力持ちと言える存

在である。

2.4.1. 潜在拡散モデルにおけるVAEの役割

高解像度の画像（例：1024x1024ピクセル）をピクセル空間で直接扱う拡散プロセスは、膨大な計算コストを必要とする。LDMは、この問題を解決するために、ボトルネック型VAEを用いて画像をより低次元の「潜在空間（Latent Space）」に圧縮し、その空間内で拡散およびデノイズ処理を行う。本書で言及するVAEは、特段の断りがない限り全て、潜在空間が入力画像空間より低次元の、ボトルネック型画像VAEを指すものとする。
具体的には、学習時および推論時に以下の2つのプロセスが発生する。

1. エンコード: VAEのエンコーダが、入力された高解像度画像を、その本質的な意味的特徴を保持したまま、低次元の潜在表現（Latent Representation）に変換する。例えば、512x512x3チャンネルの画像が、64x64x4チャンネルの潜在表現に圧縮される。
2. デコード: 推論アーキテクチャ（U-NetやDiT）によるデノイズ処理が完了した後、VAEのデコーダが、クリーンになった潜在表現を受け取り、それを元のピクセル空間の高解像度画像に復元する。

この潜在空間での操作により、推論アーキテクチャが扱うデータの次元数が劇的に削減され、計算負荷とメモリ使用量が大幅に軽減される。

2.4.2. 圧縮率とディテール保持のトレードオフ

VAEの設計、特にその「圧縮率」は、最終的な画質に決定的な影響を与える。これは、しばしば見過ごされがちだが、画像生成における最も重要なボトルネックの一つである。VAEによる圧縮は非可逆であり、エンコードの過程で一部の情報は必然的に失われる。問題は、どの程度の情報が、どのような形で失われるかである。
この点を理解するためには、まず、なぜ多くの画像生成に関する問題が、実際にはVAEに起因するのかを認識する必要がある。ユーザーがしばしば不満を抱く、生成された画像における不鮮明な顔、不自然な形状の手、判読不能な文字といった問題は、デノイズ処理の失敗ではなく、最初のVAEによる圧縮段階で、それらの微細なディテール情報がすでに失われていることに起因する場合がある。一度潜在空間に渡る前に失われた情報は、後段のいかに高性能なデノイザーでも復元することは原理的に不可能である。
この問題は、VAEのチャンネル数と圧縮率に直接関係している。Stable Diffusion 1.5やSDXLで採用されている標準的なVAEは、画像を4チャンネルの潜在空間に圧縮する。特にSDXL VAEの圧縮率は1:48と非常に高く、計算効率を優先する設計となっている。この高い圧縮率が、細かいディテールを失わせる主因であった。
このボトルネックを解消すべく、FLUXやStable Diffusion 3といった最新モデルでは、16チャンネルの潜在空間を持つ新しいVAEが導入された。例えば、FLUX VAEの圧縮率は1:12であり、SDXL VAEに比べて圧縮を大幅に緩めている。チャンネル数を増やし、圧縮率を下げることで、VAEはより多くの微細な情報を潜在空間に保持できるようになる。これにより、従来は困難であった顔の細かな表情、皮膚の質感、小さな文字などの忠実な再現が可能となった。AuraDiffusionによって再現された16チャンネルVAEの性能評価では、従来のVAEに比べてPSNR（再構成品質）やLPIPS（知覚的類似性）といった指標で大幅な改善が見られ、このアーキテクチャの優位性が定量的に示されている[16]。
したがって、VAEの進化は、単なる前処理・後処理ツールの改良ではなく、画像生成パイプライン全体の品質上限を引き上げるための根源的な改良と言える。これは、生成速度や効率性のみを追求する時代から、最終的な画質と忠実度を最優先する設計思想へのパラダイムシフトを反

映している。
基盤技術の進化を踏まえ、本章では個別の最新画像生成AIモデルを取り上げ、それぞれのアーキテクチャを「テキストエンコーダ」「画像推論アーキテクチャ」「VAE」の3つの構成要素に沿って詳細に分析する。各モデルがこれらのコンポーネントをどのように組み合わせ、どのような特徴を実現しているかを明らかにすることで、技術の最前線における具体的な実装とその設計思想を深く理解する。

2.5. Stable Diffusion 1.5

Stable Diffusion 1.5（SD1.5）[1]は、2022年にRunwayMLによってリリースされ、オープンソースの高性能画像生成AIとして広く普及した、この分野における金字塔的なモデルである。そのアーキテクチャは、後の多くのモデルの基礎となる、シンプルかつ効果的な構成を確立した。

- テキストエンコーダ: CLIP ViT-L/14[17]。OpenAIのCLIPモデルの中でも、Vision Transformer (ViT) Largeを画像エンコーダに、パッチサイズ14を使用したバージョンをテキストエンコーダとして採用している。この単一のCLIPエンコーダが、入力された英語のプロンプトを解釈し、その概念的な意味を768次元の埋め込みベクトルに変換する。このベクトルが、後段のU-Netに対する条件付けとして機能する。
- 画像推論アーキテクチャ: U-Net。潜在拡散モデルの中核をなすデノイザーとして、拡散モデルにおいて標準的な方式のU-Netアーキテクチャ[4]が採用されている。このU-Netは、VAEによって圧縮された潜在表現と、タイムステップ情報、そしてCLIPから得られたテキスト埋め込みを入力として受け取る。クロスアテンション機構を介してテキスト条件を反映させながら、反復的にノイズを予測・除去する。
- **VAE**: Stable Diffusion VAE（具体的にはvae-ft-mse-840000-ema-prunedとして知られるファインチューン版）。このVAEは、512x512ピクセルの画像を、8分の1の解像度である64x64ピクセル、4チャンネルの潜在空間に圧縮する。推論の最終段階では、生成された潜在表現を元のピクセル空間にデコードし、最終的な画像を生成する。このVAEは、後のモデルでも長らく標準として使われたが、色の彩度が低めに出る、細部がぼやけるといった傾向が指摘されていた。
- 後にVAEもOSSコミュニティによって改良が行われ、彩度の高いVAEなども学習されている。
- 事前学習にはLAIONデータセットを用いている。

特徴: SD1.5のアーキテクチャは、潜在拡散モデルの基本的な「三位一体」（CLIPエンコーダ、U-Netデノイザー、VAE）を確立した点で非常に重要である。LAION-5Bデータセットのサブセットで学習され、テキストからの画像生成（txt2img）、画像からの画像生成（img2img）、インペインティングなど、多彩な機能をオープンなライセンス（CreativeML OpenRAIL-M）の下で提供したことで、研究者からアーティストまで幅広いコミュニティに受け入れられ、爆発的なエコシステムを形成する基盤となった。

2.6. Stable Diffusion XL (SDXL) と Refiner

Stable Diffusion XL（SDXL）[18]は、SD1.5の成功を基に、生成画像の品質、解像度、プロンプト理解能力を大幅に向上させることを目指して開発された後継モデルである。そのアーキテクチャは、各コンポーネントを大規模化・高度化すると同時に、「専門家のアンサンブル」という新しいコンセプトを導入した点が特徴的である。

- テキストエンコーダ: CLIP ViT-L/14 と OpenCLIP ViT-bigG/14[19] の2つを併用。SD1.5のCLIP ViT-Lに加え、より巨大なOpenCLIP ViT-bigGを第二のテキストエンコーダとして採用した。これら2つのエンコーダからの出力は連結され、U-Netへの条件付けに用いられる。異なる特性を持つ大規模なエンコーダを組み合わせることで、より豊かでニュアンスに富んだテキスト表現を獲得し、プロンプトの微妙な指示を捉える能力が向上した。
- 画像推論アーキテクチャ: U-Net (大規模版)。SDXLのU-Netは、SD1.5のものと比較してパラメータ数が約3倍（ベースモデルで26億）に増加している。この大規模化により、モデルの表現力が高まり、ネイティブで1024x1024ピクセルという高解像度かつ高品質な画像の生成が可能となった。
- **VAE**: SDXL VAE。SD1.5と同様に4チャンネルの潜在空間を持つが、1024x1024の高解像度画像に対応するよう改良されている。ただし、コミュニティからは、このVAEが生成画像の細部（特に顔や手）にアーティファクトを生じさせることがあり、別途ファインチューンされたVAEを使用することが推奨される場合があった。
- SDXLがどのようなデータセットで学習されているかは公開されていないが、LAIONに加え、ImageNetなど複数のデータセットで学習が行われたと考えられる。

SDXLでは「専門家のアンサンブル（Ensemble of Experts）」と呼ばれる2段階の生成パイプラインを提案している。

1. ベースモデル **(Base Model)**: 上記のアーキテクチャを持つ主要なモデルで、テキストプロンプトから1024x1024解像度の潜在表現を生成する。この段階で画像の全体的な構成や主要な要素が決定される。
2. リファイナーモデル **(Refiner Model)**: ベースモデルの生成結果（ノイズがまだ少し残った状態の潜在表現）を入力として受け取り、さらにデノイズ処理を行う、ディテール追加に特化した別の小規模な潜在拡散モデルである。リファイナーは、高周波の細かなディテールや質感を描き加えることに特化しており、最終的な画像の写実性や品質を劇的に向上させる。しかし、Refinerモデルによる画像の改善は限定的であり、現在ではほとんど使われていない。

2.7. Stable Diffusion 3 (SD3)

Stable Diffusion 3[5,41]は、Stability AIがこれまでの知見を結集して開発した、同社のフラッグシップモデルである。プロンプト追従性、タイポグラフィ、画質のすべてにおいて既存のモデルを凌駕することを目指し、アーキテクチャのあらゆる要素が刷新されている。

- テキストエンコーダ: CLIP ViT-L, OpenCLIP ViT-G, T5 Version 1.1 XXL の3つを併用。2つの異なるサイズのCLIPモデルで視覚的・概念的なプロンプトを捉え、さらに47億パラメータを持つ巨大なT5-XXLエンコーダで言語構造を解析する。この3つのエンコーダからなる布陣により、より複雑なプロンプトの理解と、高度なタイポグラフィ性能を実現した。
- 画像推論アーキテクチャ: MMDiT (Multimodal Diffusion Transformer)。SD3の核心技術。DiTアーキテクチャをベースに、画像モダリティとテキストモダリティに対して、それぞれ異なる重みセットを持つTransformerを用意する。そして、アテンション機構の計算においてのみ、両者のトークンシーケンスを結合し、相互に注意を向けさせる。これにより、各モダリティは自身の表現空間の特性を保ちながら、相手の情報を深く参照することが可能になる。この双方向の情報フローが、特にテキストと画像の厳密なアライメントが求められるタイポグラフィ性能の飛躍的な向上に貢献した。また、FLUX.1と同様にRectified Flowを採用し、少ないステップ数での高速推論も実現している。

- **VAE**: New 16-channel VAE。手や顔といった、従来のモデルが苦手としてきた部位のディテール再現性を向上させるため、新たに開発された16チャンネルVAEを採用。FLUX VAEと同様の思想に基づき、圧縮による情報損失を最小限に抑え、MMDiTが生成する高品質な潜在表現を忠実に画像化する。
- データセット:非公開

特徴: SD3は、テキストエンコーダ、推論アーキテクチャ、VAEという3つのコンポーネントすべてを最新鋭の技術で刷新し、それらを協調させた統合システムとして設計されている。MMDiTによるテキストと画像の深い統合、3つの強力なテキストエンコーダによる言語理解の深化、Rectified Flowによる高速化、そして16チャンネルVAEによるディテール保持能力の向上により、総合力で新たな業界標準を確立したモデルである。モデルサイズは8億から80億パラメータまで複数用意され、スケーラビリティも確保されている。

2.8. Stable Diffusion 3.5 (SD3.5)

Stable Diffusion 3.5[42]は、SD3のアーキテクチャをベースに、さらなる改良と最適化を施した最新のリファイン版である。基本的な構成要素はSD3を踏襲しつつ、訓練の安定性や性能を向上させるための技術的な改善が加えられている。

- テキストエンコーダ: SD3と同様に CLIP ViT-L, OpenCLIP ViT-G, T5-XXL の3つを併用する。プロンプト理解の根幹をなす強力なエンコーダ群は維持されている。
- 画像推論アーキテクチャ: MMDiT-X。SD3のMMDiTを改良したバージョン。"X"は "improvements"を意味し、具体的な改善点として、(1) 訓練の安定性を向上させるためのQK-normalization[46]の導入、(2) Transformerブロックの初期12層におけるdual attention blocksの採用などが挙げられる。これらの改良により、モデルの全体的な性能とコヒーレンスが向上している。
- **VAE**: SD3と同様の高性能な 16-channel VAE を使用する。
- データセット:非公開

特徴: SD3.5は、SD3という強力な基盤の上に、さらなる安定性と性能向上を目指した堅実なアップデート版と位置づけられる。アーキテクチャの根幹は共有しつつ、より洗練された訓練技術や細部の改良を施すことで、完成度を高めている。訓練手法にも工夫が凝らされており、256x256から1440x1440まで解像度を段階的に上げていくプログレッシブトレーニングや、多様な解像度の画像を混合して学習させることで、様々なアスペクト比での生成能力を高めている。

2.9. FLUX.1

FLUX.1[20, 52]は、Stable Diffusionの開発者たちが設立したBlack Forest Labsによって開発された、次世代のテキスト画像生成モデルである。FLUX.1は、アーキテクチャのあらゆる側面で最先端の技術を統合しており、特に推論速度、プロンプト追従性、ディテール再現性において既存のモデルを凌駕することを目指している。

- テキストエンコーダ: CLIP と T5 の2つを併用。このハイブリッドアプローチは、SD3と同様の思想に基づいている。CLIPエンコーダがプロンプトの視覚的・概念的な側面（例:「写真風」「猫」）を捉え、巨大なT5エンコーダがプロンプトの言語的・構造的な側面（例:複雑な構文、空間関係、正確なスペリング）を精密に解釈する。ユーザーはT5用とCLIP用に別々のプロンプトを与えることも可能で、より高度な制御を実現する。

- 画像推論アーキテクチャ: Rectified Flow Transformer (120億パラメータ)。FLUX.1は、従来の拡散モデルの確率的微分方程式（SDE）に基づくアプローチから、より決定論的な常微分方程式（ODE）に基づくRectified Flow（整流化フロー）へと移行した。これにより、データとノイズ間の軌道が直線化され、理論上は1ステップでの推論も可能になるなど、推論に必要なステップ数が劇的に削減される。バックボーンには、SD3のMMDiTに類似した、マルチモーダルな入力を扱う巨大なTransformer（12Bパラメータ）が採用されていると推測されている。
- **VAE**: FLUX VAE (16-channel)。FLUX.1は、SDXL VAEの4チャンネルから大幅に拡張された16チャンネルの潜在空間を持つVAEを導入した。これにより、エンコード時の圧縮率が低く抑えられ（例：1:12）、従来は失われがちだった微細な文字、顔の表情、布地の質感といった高周波ディテールを潜在空間に保持することが可能になった。
- データセット:非公開

特徴: FLUX.1は、(1) Rectified Flowによるより高速な推論と、専用の蒸留手法による更なる高速化（Schnell版では1〜4ステップでの生成が可能）、(2) CLIP+T5による極めて高いプロンプト追従性とタイポグラフィ能力、(3) 16チャンネルVAEによる優れたディテール再現性、という3つの最先端技術を統合したモデルである。これらの組み合わせにより、品質と速度の両面で新たな基準を打ち立てた。さらに、Kontextモデルでは画像とテキストの両方をプロンプトとして入力でき、文脈に応じた画像編集が可能になるなど、マルチモーダルな対話性も追求している。

2.10. Lumina-Image 2.0

Lumina-Image 2.0[14]は、Alpha-VLLMによって開発された、アーキテクチャの「統一性」と「効率性」を核とする先進的なテキスト画像生成フレームワークである。従来のクロスアテンションによる条件付けの限界を克服し、より深いレベルでのマルチモーダル融合を目指している点が最大の特徴である。

- テキストエンコーダ: Gemma-2-2B。Googleの高性能なオープンソースLLMであるGemma-2-2Bをテキストエンコーダとして採用している。これにより、高度な言語理解能力をオープンなエコシステムの中で実現している。
- 画像推論アーキテクチャ: Unified Next-DiT。Lumina-Image 2.0の核心をなすアーキテクチャ。従来のモデルがテキスト埋め込みをクロスアテンションを介して外部から注入するのに対し、Unified Next-DiTは、テキストトークンと画像（潜在）トークンを単一のシーケンスへと連結する。この統一されたシーケンス全体に対して
  Joint Self-Attention（共同自己注意機構）を適用することで、テキストと画像の情報が区別なく、完全に相互作用することを可能にする。これは、近年のLLMにおけるデコーダオンリーTransformerのように、モダリティ間の境界を取り払う設計思想であり、より自然で双方向的な情報交換を促進する。
- **VAE**: FLUX-VAE-16CH。ディテールの再現性に定評のある、FLUX.1と同様の高性能な16チャンネルVAEを採用している。これにより、Unified Next-DiTが生成した高品質な潜在表現を、情報を損なうことなくピクセル空間に復元する。
- データセット:非公開

特徴: Lumina-Image 2.0のアーキテクチャは、「統一性」という思想に貫かれている。テキストと画像を分離されたモダリティとして扱うのではなく、初めから一つの連続した情報ストリームとして扱うことで、従来のクロスアテンション機構が持つ一方向的なバイアスの問題を回避し、より深いレベルでのマルチモーダルな理解と生成を目指す。この統一アーキテクチャは、将来的に他のモダリティ（音声、動画など）のトークンを追加する際にも、アーキテクチャの根本的な変更なしに拡

張できるというスケーラビリティも備えている。

2.11. 既存モデルの問題点と新規モデルに要求されるスペック

2025年9月の段階においては、Stable Diffusion XLが多くの支持を得ておりユーザー数が最も多い。これはライセンスと動作スペックの問題が大きく、最新のモデルの殆どが商用利用不可などの制限がつき、家庭用のPCでは扱いにくいモデルサイズであることが普及しない要因となっている。

今回のプロジェクトで対象とするアニメ事業者の立場から各モデルを評価するときに、データセットの問題が立ちはだかる。Stable Diffusion 1.5 は複数の倫理的な懸念があるLAIONデータセットを用いており、他のモデルに関してはどのようなデータセットで学習が行われたのかを公開していない。このようなモデルをアニメ制作に用いることはリスクが高く、使用をためらわせる原因になっている。

新しいモデルにはこれらの問題をクリアすることが要求される。すなわち、学習されたデータセットが公開されていること、データセットの著作権問題がクリアされていることである。しかし、著作権問題がクリアされたデータセットはLAION（数億枚）に比べて画像の数が少ない。生成モデルを学習するにあたってデータの少なさはモデルの性能を左右することから、少ないデータでも性能を発揮できるモデルの開発が必要となる。

また、著作権問題がクリアされたデータセットの中に著作権の残っているデータが混ざっていることが後から発覚することがある。一度学習してしまうと一部のデータのみを学習内容から除去することは難しく、基盤モデルを最初から学習する必要が出てくる。よって、少ない学習量で出力が行えるようになる基盤モデルが必要となる。

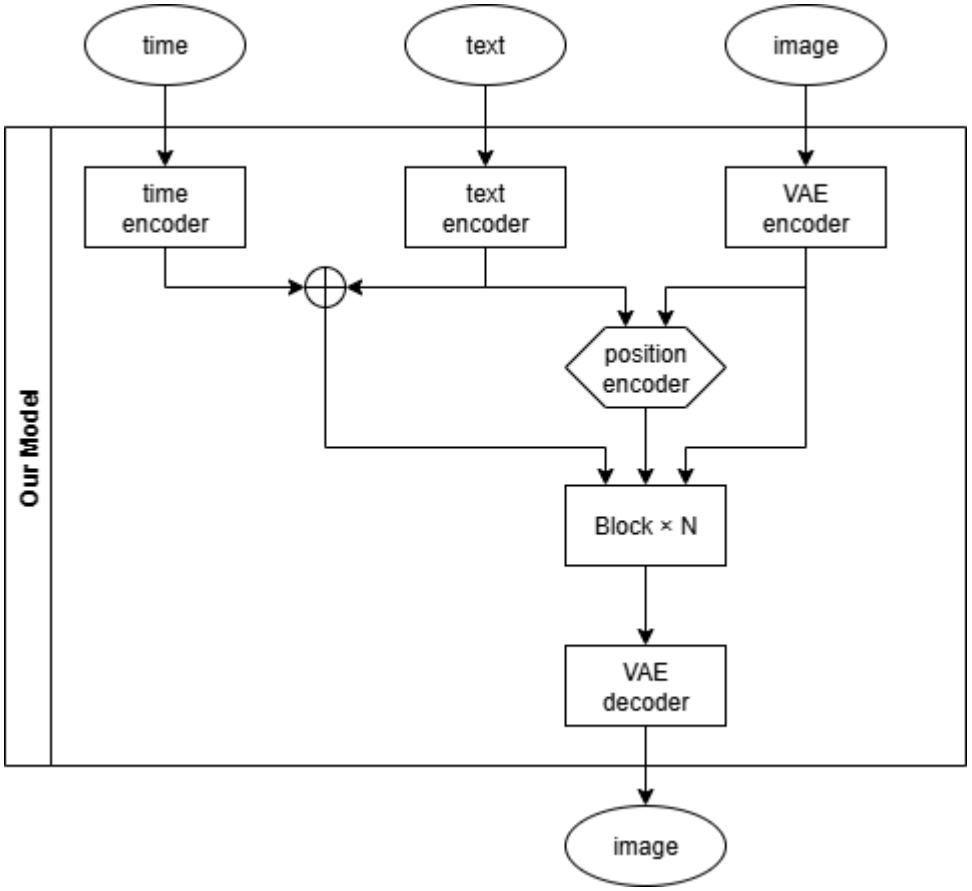
3. 開発モデル

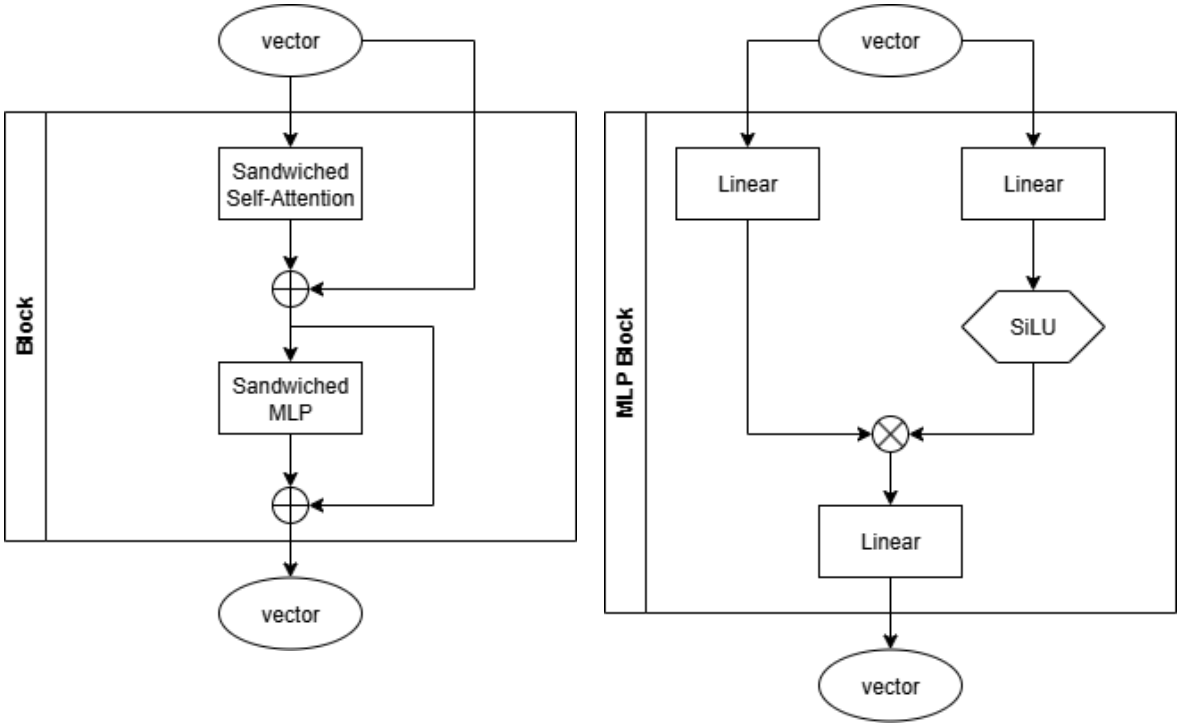
3.1. モデルの全体構成

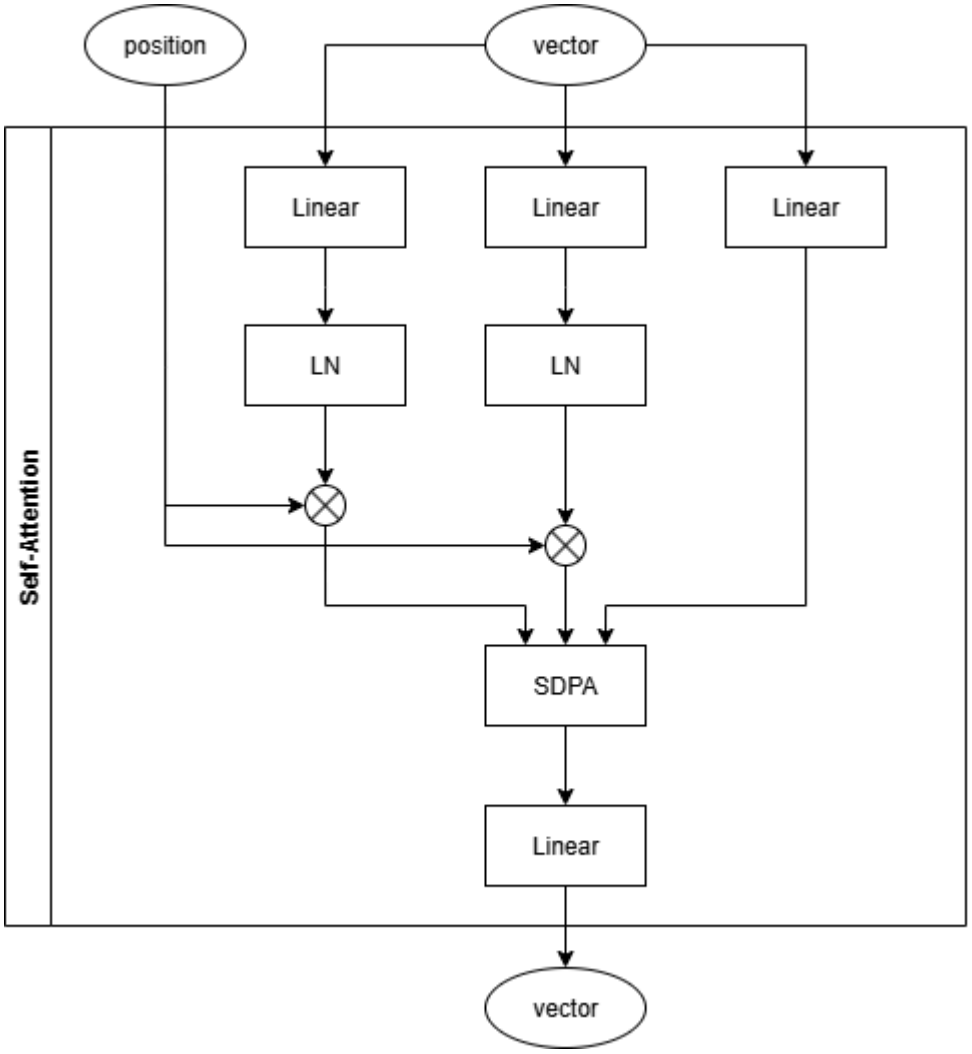
oboro:は、テキスト条件付けを利用した画像生成拡散モデルであり、I-CFMの流れ定式化を用い、3.2.にて示すDiT構造とMulti-Multi-Head Attention構造を採用したモデルである。また、テキストエンコーダにはT5 V1.1 XXLを採用し、VAEにはFLUX VAEを採用した。

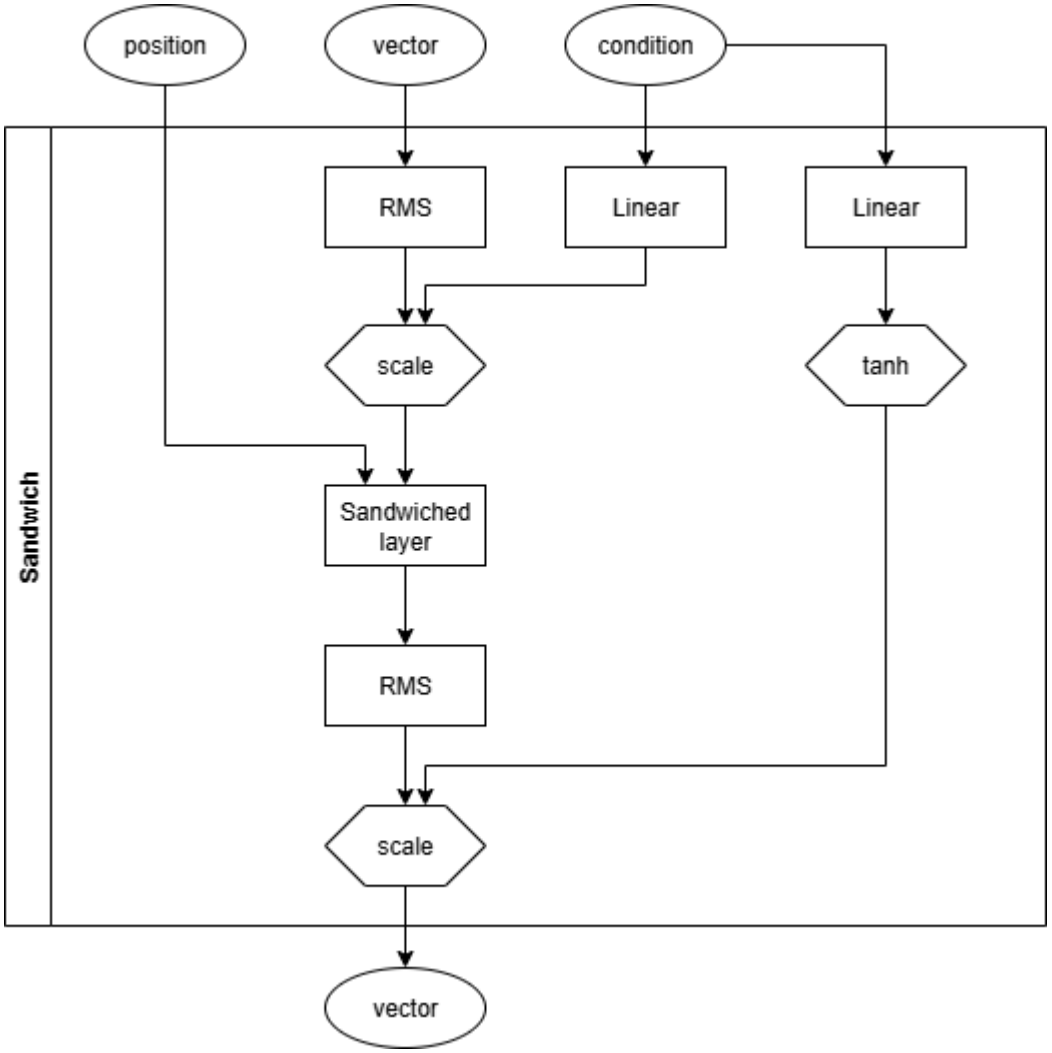
3.2. ネットワーク構造の詳細

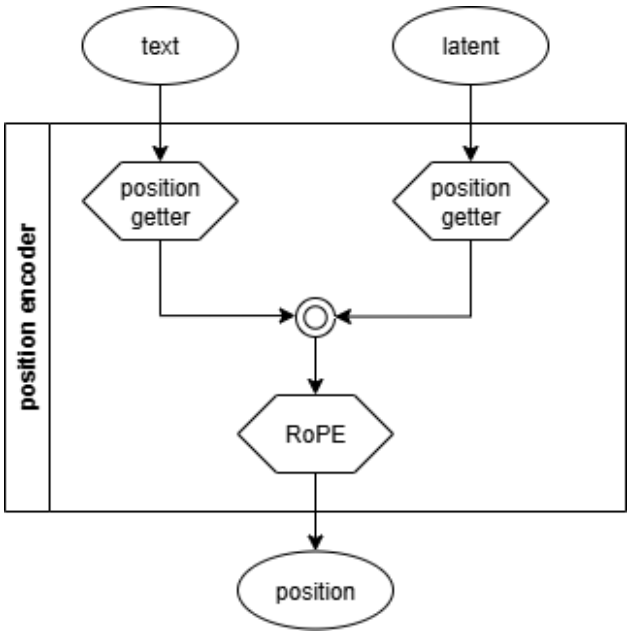
開発したモデルのネットワーク構造の詳細図は以下の通りである。始めに我々のモデルの全体構造を示し、その後、各ブロック内の構造の詳細図を示している。




vector
Block
Sandwiched Self-Attention
Sandwiched MLP
vector
vector
MLP Block
Linear
Linear
SiLU
Linear
vector


position
vector
Self-Attention
Linear
Linear
Linear
LN
LN
SDPA
Linear
vector


position
vector
condition
Sandwich
RMS
Linear
Linear
scale
tanh
Sandwiched layer
RMS
scale
vector


text
latent
position encoder
position getter
position getter
RoPE
position

3.2.1. DiffusionTransformer (DiT)

本研究では、Stable Diffusion 3, FLUX.1, Lumina-Next-T2I[21], Lumina-Image 2.0, Mochi[22]の方式に近いDiT構造を採用した。Encoder部では、時刻、入力文章、入力文章のBOS（beginning of sequence）トークン、入力画像、トークン位置ID表が、それぞれ3層MLPまたは全結合層によって埋め込み表現へとエンコードされる。そしてTransformer部では、SwiGLU非線形性[22], 2D-RoPE[23,43], Sandwich正規化[44,49], QK正規化[46], Joint Self-Attention[14]を用いる構造である。

3.2.2. Multi-Multi-Head Attention

本研究で採用した複数層の DiT (Diffusion Transformer) ブロックから成る生成モデルアーキテクチャには、Multi-Multi-Head Attention と呼ぶ新規な設計を導入した。これは、各DiTブロック内のアテンション層において、アテンションヘッド数を層ごとに異なる値に設定する手法である。

この設計は、画像セグメンテーションなどで広く用いられる U-Net アーキテクチャの階層的な特徴処理から着想を得ている。単純に同一構造のDiTブロックを複数層重ねた場合、各ブロックが担うべき役割の分化がうまく進まず、特に学習の初期段階において特徴抽出の効率が低下する懸念がある。例えば、全ブロックが画像の細部と全体構造の両方を同時に学習しようとすると、学習の収束が遅くなる可能性が考えられる。U-Netにおいては棲み分けがよくできており、層ごとに異なる要素の学習が行われていた。

この課題に対処するため、Multi-Multi-Head Attentionは、アテンションヘッド数を調整することで、各DiTブロックに暗黙的な役割分担を促すことを目的とした。

U-Netが前半の層で大域的な特徴を捉え、後半の層で局所的な特徴を復元していくように、本手法ではアテンションヘッド数によって情報の粒度を制御する。各層のhead数をどう配分するかについては層数を少なくした検証実験により最適な値を求めることに成功した。結果、徐々に層数を増やしていく形式において、学習効率の向上が確認された。

- 初期のDiTブロック（少ないヘッド数）: ヘッド数を少なく設定する（例：8,16）。各ヘッドが担当する部分空間が広くなるため、モデルは画像のレイアウトや物体間の関係性といった、より大域的で低周波な特徴を捉えるよう促される。
- 後段のDiTブロック（多いヘッド数）: ヘッド数を多く設定する（例：24, 48）。各ヘッドはより専門化し、狭い部分空間を担当する。これにより、テクスチャや輪郭といった、局所的で高周波な詳細を精緻に生成する役割を担わさせる。
- 実験ではhead数を8,16,24,48とする方式が良かったため、本学習においても採用している。

3.3. 学習戦略と損失関数

モデルの学習における学習戦略として、I-CFM (Independent Conditional Flow Matching) を採用し、損失関数としてMSE（mean squared error）を採用した。この手法は、データサンプル x_1 とノイズサンプル x_0 を直線的に補間する経路 $x_t = tx_1 + (1 - t)x_0$ 上の流れを学習する Rectified Flow[26] のアプローチに類似している。

Rectified Flow のような流れの学習を効率化する手法として、最適輸送 (Optimal Transport) の概念を導入した OTCFM[27] が提案されている。OTCFM はデータ分布とノイズ分布間の最適な対応付けを用いて学習を行うことで離散化誤差を大幅に減少させ、数ステップの高速推論を実現する手法であるが、テキストによる条件付けと組み合わせた我々の小規模実験においては十分な性能を発揮できなかった。そこで、同様にOT-CFM論文にて提案されたI-CFMを採用した。

I-CFMは、Rectified Flowの流れに対して推論をより頑健にするような雑音項を追加したのと等価な手法であり、Macha-TTS[25]のような採用例がある。

3.4. テキストエンコーダの選定

テキストエンコーダはGemma-2 2B, Gemma-2 2B (rinna-FT), Gemma-2 9B, Quen-2 1.5B, Quen-2 3B, Quen-2-VL 2B,Phi-3.5-mini-inst, T5-1.1-XXL, CLIP-ViT L/14, SIGLIP[45], SIGLIP-so400Mを候補としたが、最終的にT5 Version 1.1 XXL[11,28]を採用した。これは、小規模実験の結果、より入力文章を反映した画像が生成されたためである。

4. 学習プロセス

4.1. データセット

4.1.1. 選定

本プロジェクトにおいてはアニメ製作補助を念頭に置いており、ユーザー企業が安心して使用できるモデルとして開発する必要がある。このとき著作権に配慮されたデータセットを用いることが求められる。2章で述べたとおり、ほとんどの競合モデルは無許諾のデータセットで学習が行われているか、データセットを公開していない。アニメ事業者がビジネス上で利用可能なモデルとするため著作権に配慮されたデータセットを選定する。

この観点から、我々はMegalith-10mデータセット[29]を選定した。このデータセットが選ばれた主な理由は、その構築プロセスにおいて明確に著作権に配慮したデータソースのみが使用されている点にある。内容としては写真共有サイトFlickerにおいてCC0相当で公開されている画像を集めたデータセットであり、比較的大規模で多様な画像が利用できる。

4.1.2. 重複除去

Megalith-10mにおいては重複する画像が多くあったため、重複画像の除去を行った。似たような画像が多くあると過学習や学習が不安定になったり、出力画像の偏りが生じる恐れがあるためである。

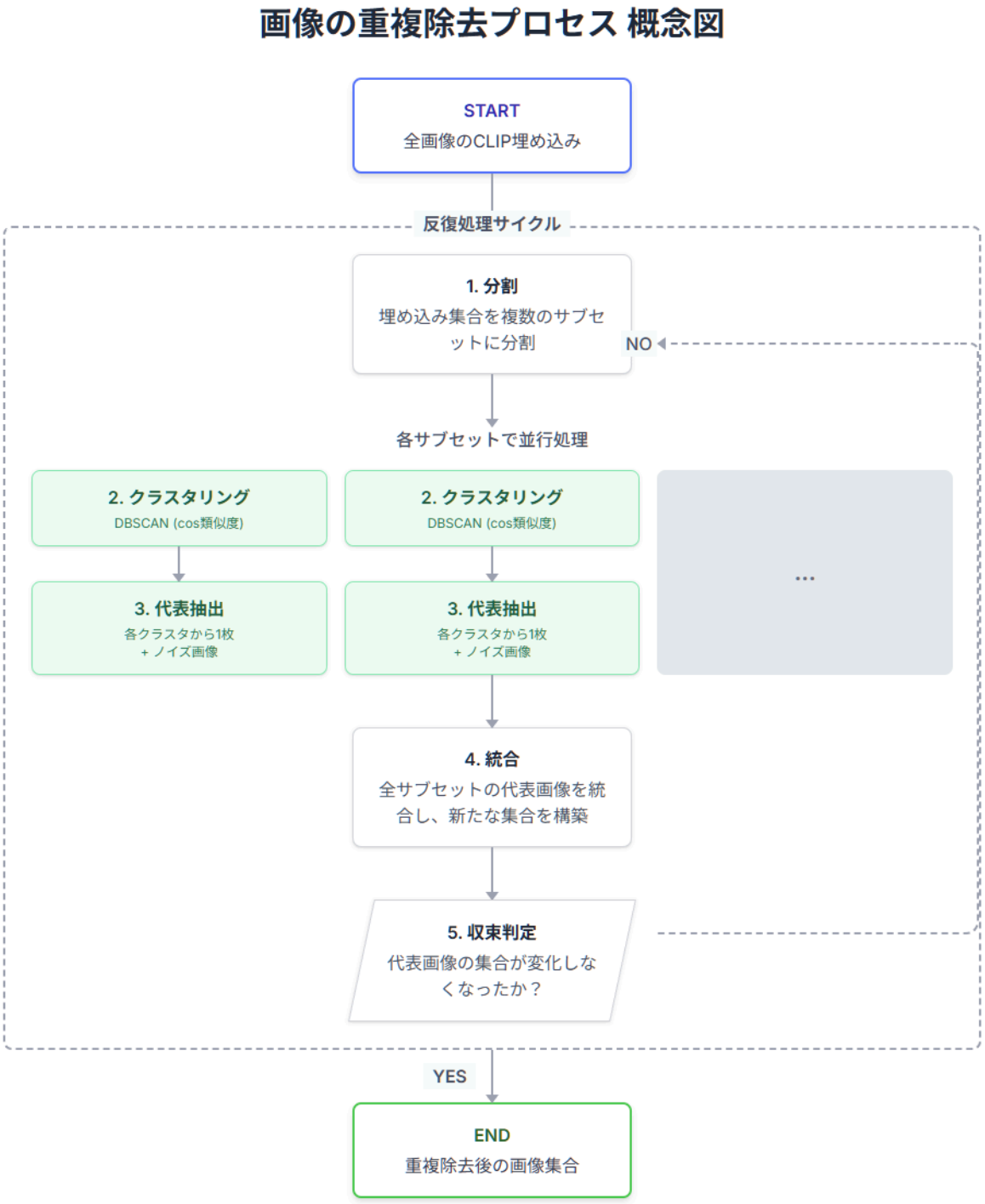
データセット内に含まれる重複、あるいは極めて類似性の高い画像を効率的に除去するため、CLIPモデルによって得られる画像埋め込みベクトルを用いた反復的クラスタリング手法を実装した。処理の概要は以下の通りである。

まず、データセット内の全画像に対し、事前学習済みモデルCLIP (DataComp版 L/14)を適用し、それぞれの画像埋め込みを取得した。

次に、計算負荷を軽減するため、全埋め込みベクトルの集合を数個のサブセットに分割した。各サブセットに対し、DBSCAN法によるクラスタリングを実行した。サンプル間の計量にはコサイン類似度を用い、DBSCANの類似度閾値（eps）は0.9に設定した。この閾値は、類似した画像が同一クラスタに分類されるよう経験的に調整された値である。

クラスタリング後、各サブセットについて、「形成された各クラスタから1枚の画像をランダムに抽出したもの」と、「いずれのクラスタにも所属しなかったノイズ画像」を合わせたものを、そのサブセットにおける代表画像の集合とした。

こうして全サブセットから選出された代表画像をすべて統合し、新たな画像集合を構築する。この集合を次の処理対象とし、再びサブセットへの分割から代表画像の選出までの一連のサイクルを繰り返した。この反復プロセスを、選出される代表画像の集合が収束し、ほぼ変化が見られなくなるまで実行することで、データセット全体の重複を網羅的に除去した。



4.1.3. キャプショニング

Megalith-10mは画像のデータセットとなっており、キャプションテキストがShareCaptioner[30]など複数公開されている。本研究では複数種類のタグ付与を行うため、VLMの選定とキャプション作成を独自に行った。

本プロジェクトではFlorence-2, Molmo, WD 1.5 Tagger, Pixtral, Phi-3.5 vision instruct, Qwenなど複数のVLMを用いて比較検討を行った。結果として、Florence-2[31]を採用した。これは他のVLMと比較してアニメ画像の表現理解が優れていたことと、生成速度が大幅に高速であったこと、Florence-2の機能で安定して複数の長さのキャプションを生成し分けることができ、目的に合

致したことによる。約1000万枚の画像にキャプション付けする必要があったため、速度を重視した。WD 1.5 Taggerについては生成速度は問題ないものの、生成されるキャプションが利用用途に向かないタグの羅列として出力されてしまうので採用しなかった。

4.1.4. スコアリング

各画像について画像のクオリティを表す専用タグを設けてキャプションに追加している。これはデータセットの画像を画像のクオリティにより複数のカテゴリにわけ、専用のタグを加えることで、画像生成時に高品質のタグをつけて生成することで生成画像の品質を向上させる手法である。逆に、ネガティブプロンプトに低品質のタグを入れて品質を向上させる事もできる。

画像のスコアリングには国内開発者により作成され、画像生成モデル製作者からも高く評価されているAesthetic Predictor V2.5[32]を用いた。これはSigLIPベースの予測ツールであり、画像の品質・美的スコアを1から10のスケールで評価する。スコアごとにタグの追加を行っており、10～6がexcellent score（上位約3%）、6～5.2がgood score（上位約20%）、5.2～4（上位約80%）をaverage scoreとした。スコアが4以下の画像についてはデータセットから除外した。除外された画像は全体の20%程度となる。除外の閾値は画像をランダムにピックアップして目視で決定した。

4.1.5. Latent化

画像をVAEを通してLatentに変換し学習を行う。学習時にVAEの変換を行うと学習時間が余分に使われてしまうため、事前にすべての画像をLatentに変換して学習を行った。これにより、VAEエンコードの繰り返し計算を省略して計算量を削減し、さらにメインメモリからVRAMへの転送量を低減できた。この作業はローカル環境でも行えるため学習テストと並行してPCで行っている。この作業により、結果として学習速度が向上し、より効率的な学習ができた。

4.2. 学習環境

4.2.1. ハードウェア

本プロジェクトではAWS ParallelClusterのH100x8環境を用いた。利用したAWS ParallelClusterとジョブスケジューラSlurmによる計算クラスター環境において、安定性に関する課題が散見された。数時間から数日おきという不規則な間隔で、一部の GPU プロセスが停止する事象が確認された。再現検証の結果、学習コード単独では説明がつかない可能性があると判断し、ベンダーの技術支援を得ながら原因の切り分けと対策を進めたが、本フェーズ中に恒久対策の確立には至らなかった。なお、同様のハードウェア構成に関する外部の報告も見られる[33]

当該事象は学習スループットおよびスケジュールに影響を与え、約 8 割程度の実効効率を維持するために時間帯を問わない復旧対応が必要となる局面があった。これにより運用チームの負荷が増大し、プロジェクトマネジメント上のリスク要因となった。

本件の知見を踏まえ、同様の構成を採用する場合には、事前検証と冗長化を前提とし、安定した計算基盤の確保を優先する必要があるとの示唆が得られた。

4.2.2. ソフトウェア

作業環境構築

本プロジェクトにおける開発環境の構築には、Pythonの統一的な管理ツールであるRye、およびその内部で利用される高速なパッケージインストーラであるuvを採用した。

Rye

Rye[47]は、Pythonのバージョン管理、仮想環境の構築、依存パッケージの管理といった、Pythonプロジェクトで必要とされる一連の機能を単一のコマンドラインインターフェースで提供するツールである。従来、これらの作業はpyenv, venv, pipといった複数のツールを個別に利用する必要があり、手順が煩雑化する一因となっていた。本プロジェクトでは、Ryeを導入することでこれらのツール群を一つに集約し、環境構築手順を標準化・簡素化することを目的とした。

uv

uv[48]は、Rust言語で実装された軽量なPythonパッケージインストーラおよびリゾルバである。従来のpipと比較して、依存関係の解決やパッケージのインストールにかかる時間を劇的に短縮する能力を持つ。Ryeは標準でuvを内部的に利用するため、Ryeを介したパッケージ操作は自動的にuvの速度的な恩恵を受ける。高速なパッケージ操作は、ライブラリの試用や環境の再構築を頻繁に行う現代的な開発プロセスにおいて不可欠であると判断し、採用の決め手とした。

導入による効果

RyeおよびuvのRyeの導入により、作業環境の構築効率は著しく向上した。チームメンバーは単一のryeコマンドを学習するだけで、Pythonバージョンの指定から依存関係のインストールまでを一貫した手順で実行可能となった。これにより、セットアップにかかる時間と認知負荷が大幅に削減された。

特に、uvの高速なパッケージインストール能力は顕著な効果を発揮した。大規模なライブラリや多数の依存関係を持つパッケージを追加・更新する際の待機時間が大幅に短縮され、開発サイクルが高速化した。結論として、これらのツールの採用は、プロジェクトのスムーズな立ち上げと、開発者間での環境差異の抑制に大きく寄与し、開発全体の生産性を高める上で非常に有効であった。

PyTorch Lightning

本プロジェクトでは、機械学習フレームワークとしてPyTorchを高レベルで抽象化するPyTorch Lightningを採用した。これにより、定型的なコードを削減し、研究開発の初期段階における実装速度の向上が図られた。

しかしながら、開発を進める中でいくつかの課題が浮上した。特に、PyTorch Lightningが内部で利用するPyTorch Distributed由来のデータ集約および共有の処理において、実装上の困難に直面する場面が多かった。より細かな分散学習制御や設定のカスタマイズが求められる局面では、PyTorch Lightningの抽象化が逆に制約となる可能性が示唆された。PyTorch Lightningでは内部でどのような処理が行われているかが見えにくく、問題が発生したときのデバッグが難航することがあった。

この経験から、プロジェクトの要件によっては、PyTorch Distributedを単独で採用し、より低レベルでの制御を行うアプローチの方が適していた可能性も考えられる。今後の開発においては、これらの分散学習処理における問題への対策を講じる必要がある。

Weights & Biases (W&B)

モデルの学習過程をモニターし、実験管理を行う目的でWeights & Biases (W&B) を導入した。これにより、損失や精度といった各種メトリクスの可視化、ハイパーパラメータの追跡、生成画像の確認が効率的に行えるようになり、実験の再現性と分析の精度が向上した。

4.3. ハイパーパラメータ

4.3.1. アーキテクチャ

層数を調整可能なTransformer層は32層とした。様々な層数で性能を試したところ、32層までは層数を増やすことで性能が向上することが確認できた。32層以上にすることも検討したが、VRAMやローカルでの扱いやすさなどを考慮して32層となった。

学習対象外の層

エンコーダ部のうち、学習済みのTEであるT5 V1.1 XXLと学習済みのVAEであるFLUX VAEと、学習不要なRoPEエンコーダ。

学習対象の層

エンコーダ部のうち凍結されていない層と、Transformer中間層32層およびTransformer最終層。

パッチサイズ

画像をpixel unshuffle[34]する際の縮小率係数をパッチサイズ (patch size) と言う[5,20]。計算量を鑑みた結果、SD 3やFLUX.1と同様に、パッチサイズは2に設定している。

4.3.2. オプティマイザ

本プロジェクトの学習プロセス全体を通して、オプティマイザにはAdamW[35]、スケジューラにはウォームアップ（warmup）付きのものを一貫して採用した。この組み合わせは、近年の大規模モデルの学習において広く採用されており、安定した学習結果が得られる実績が豊富である。本プロジェクトでは、学習の失敗リスクを極力排除し、安定性を最優先事項としたため、他の新規性のある候補の採用は見送った。

検討段階では、メモリ効率の良いAdafactorやLionといったオプティマイザの利用も候補に挙がった。しかし、本プロジェクトの計算環境においてはVRAMに十分な余裕があったため、メモリ使用量を削減するメリットは限定的であると判断し、採用には至らなかった。また、RAdamScheduleFreeも試用したが、標準的なAdamWとウォームアップの組み合わせと比較して、特に顕著な優位性は確認できなかった。

以上の検討を経て、実績があり安定しているAdamWとウォームアップ付きスケジューラの組み合わせが、本プロジェクトの目的達成に最も適していると結論付けた。

オプティマイザのパラメーターβ_1, β_2, epsは0.9, 0.999, 1e-8とした。

4.3.3. 学習率

256x256解像度事前学習段階（4.4.2にて後述）での学習率は、2e-4とした。512x512面積追加事前学習段階での学習率は1e-4から始め、7e-5に減少させた。追加学習段階では、2e-5から始め1e-6まで学習率を漸減させた。これらは実際に学習を走らせながら様子を見て決定した。

4.3.4. バッチサイズ

256x256解像度段階では、合計バッチサイズ1024（2ノード、ノード内GPU数8、GPU内バッチサイズ64の総乗）にて学習した。512x512面積段階では、合計バッチサイズ384（2ノード、ノード内GPU数8、GPU内バッチサイズ24の総乗）にて学習した。

4.4. 学習の実行

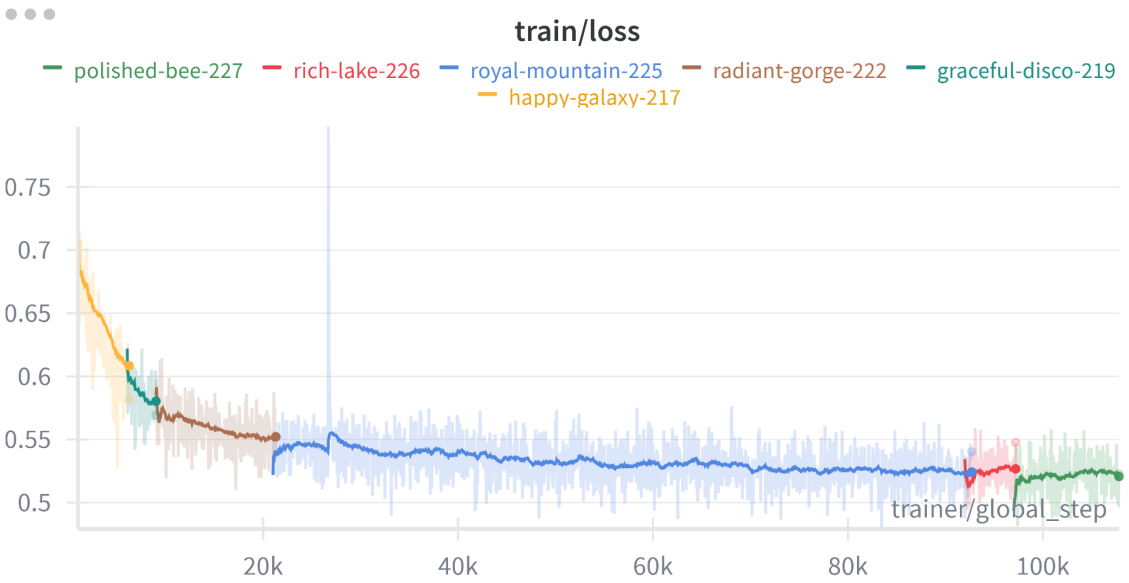
4.4.1. 小規模学習試験

本プロジェクトではDiT (Diffusion Transformer) アーキテクチャを採用している。このアーキテクチャの利点はDiTの層数を変更することでモデルのサイズを調整できる点である。本格的な学習に先立ち、計算リソースを効率的に活用するための予備検証として、小規模な学習試験を実施した。DiTのTransformerブロック数を削減した軽量モデル（20層）を構築し、これを用いて学習を行った。この措置により、学習パイプライン全体の動作確認やハイパーパラメータの初期探索を、少ない計算コストで迅速に行うことを目的とした。

4.4.2. 段階的な学習

事前学習は256x256解像度正方形画像での学習と512x512面積長方形画像（およそ250kピクセルの画像）の学習の2段階に分けて行った。また、追加学習は512x512面積長方形画像で行った。

4.4.3 学習の収束過程



上図は、W&Bで記録した、256x256解像度での事前学習の学習損失である。図中には、実際のデータ値を薄い線、時間加重指数移動平均値(係数0.9)を濃い線で示している。
royal-mountain-225の学習序盤のloss spikeでは目視結果で問題が見受けられなかったため学習を継続した。図に見られる学習中断は、4.2.1.に記述のハードウェア問題によるものである。

4.4.4 学習を安定させるために行った工夫

256x256解像度での事前学習段階では安定して学習が進んでいたが、512x512面積長方形画像での中間学習段階やアニメ特化追加学習段階では度々損失スパイクが発生し学習が不安定になった。このとき、学習率を手動で漸減することにより、問題に対処した。Cosine減衰のような決め打ちスケジューラでは安定しなかったため、将来的にはScheduleFree[24]などの安定運用について模索したい。また、ZClip[23]は安定化に寄与しなかった。損失スパイクについての理論は様々なものが提案されており、今後の発展が期待される。

4.5. コミュニケーションと情報共有

プロジェクトにおけるメンバー間の意思疎通および情報共有は、主にコミュニケーションプラットフォームであるDiscordを中心に行った。

日常的なコミュニケーションの場に、記録やメモといったフロー情報を一元化することが、運用の観点から最も効率的であると判断したためだ。Discord上では数日おきの定期打ち合わせの他、日常から気になる論文などを共有したり、議論を深めるなどコミュニケーションの場として活用した。一方で、各自の調査研究といったストック型の知識はNotionで報告書として管理し、プロジェクトの公式な報告書はGoogle Docsで共同編集するなど、情報の性質に応じてツールを使い分ける方針をとった。

また、関連情報へのアクセス性を高めるため、Hugging FaceのリポジトリやW&Bの実験ページへのリンクをDiscordチャンネルに紐付けた。学習の進捗やエラーを自動で通知するDiscord botを開発・導入し、プロジェクト状況のリアルタイムな把握に努めた。

5. 実験と評価

5.1. 生成結果の定性的評価

図:「**oboro:base**」の出力例

開発したモデルは「oboro:base」という名前をつけた。「oboro:」までが共通の名前であり、各ユーザー企業ごとに特化した学習を行った際には「oboro:ユーザー企業名」のように名前をつける事を想定している。

図は「oboro:base」は追加学習を行っていない基盤モデルであり、実写画像のみで学習されているため実写の画像が生成できる。図には下記のプロンプトを入力している。

None

A serene spring landscape outdoors, featuring abundant cherry blossom trees with delicate pink flowers and falling petals. A large body of reflective water, such as a lake or river, is in the foreground, with a strong focus on the clear reflections of the surrounding scene and sky on its surface. A picturesque wooden bridge with a railing spans the water or crosses a path nearby. A path leads through lush green grass and bushes along the riverbank or shore. A dense forest covers rolling hills, and majestic mountains form a dramatic mountainous horizon in the background under a cloudy sky. No humans are presented. best quality, highly detailed.

自然な画像が生成できており、開発・学習は成功を収めたといえる。

5.2. 評価指標

5.2.1. FID (Fréchet Inception Distance)

FIDスコア[36]は、生成された画像と実際の（本物の）画像の分布の類似性を測定する指標である。生成された画像と実際の画像との分布の類似性を測定する。Inceptionモデルという事前学習済みの画像分類モデルを用いて、両方の画像セットから特徴量を抽出し、それぞれの特徴量分布を多次元ガウス分布としてモデル化する。この2つのガウス分布（生成画像の特徴量分布と実画像の特徴量分布）の間のフレシェ距離（Fréchet distance）を計算したものがFIDである。この距離が小さいほど、生成された画像の分布が実際の画像の分布に近い、つまり生成モデルの性能が高いと判断される。FIDスコアが小さい数値であるほど、生成された画像はよりリアルで、本物の画像セットの多様性を正確に反映しているとみなされる。

5.2.2. CLIP Score (CLIP類似度)

CLIPスコア[37]は、生成された画像が、その画像を生成するために与えられたテキストプロンプトにどれだけ合致しているかを評価する指標。画像とテキスト間の類似度を測定するCLIP (Contrastive Language-Image Pre-training) モデルを使用して計算される。CLIPスコアが高いほど、生成された画像はテキストの記述内容をより忠実に表現していると判断される。

5.2.3. GenEval (Generative Evaluation)

GenEval[38]は、生成モデルの全体的な品質と多様性を評価するための総合的な指標群を指す。具体的な計算方法は文脈によって異なる場合があるが、一般的には生成画像の品質、多様性、プロンプトへの追従性など、複数の側面を考慮して評価される。低いほど良い場合と高いほど良い場合があり、具体的なGenEvalの定義に依存する。

5.2.4. TIFA Score (Text-to-Image Fidelity and Alignment)

TIFA Score[39]は生成された画像がテキストプロンプトにどれだけ「忠実（Fidelity）」で「整列（Alignment）」しているかを評価する指標。特に、プロンプトに含まれる複数の要素（オブジェクト、属性、関係性など）が画像内で適切に表現されているかを詳細に評価するために使用される。TIFAスコアが高いほど、テキストプロンプトの意図をより正確に反映した画像を生成できていると判断される。現在のテキスト画像生成モデルにおいては、プロンプトに対する正確性が良くないためにシードを変えて複数の画像を生成し、その中からプロンプトに忠実な画像を選別する必要がある。これはアニメ製作現場において余分な工数を招き、本来の目的である製作現場の負担軽減が達成できない。よって、TIFAスコアは今回測定する指標の中でも重要な指標である。

5.2.5. Aesthetic Score (美的スコア)

Aesthetic Score[40]は、生成された画像が視覚的にどれだけ「美しい」または「魅力的」であるかを評価する指標。人間の美的感覚に基づいてスコア付けされたデータセットで学習されたモデルによって推定されることが一般的である。美的スコアが高いほど、生成された画像はより人間に好まれやすい、視覚的に心地よいものと評価される。

5.2.6. Win Rate (勝率)

Win Rateは、特に人間による評価（Human Preference）を用いる指標。複数のモデルが生成した画像を比較し、どちらの画像がより優れているかを人間に投票してもらい、その投票結果に基づいて計算される。勝率が高いほどそのモデルが生成した画像が他のモデルの画像よりも人間に好まれる傾向が強いことを意味する。本プロジェクトでは生成画像と入力したプロンプト両方を表示して、画像の品質・プロンプト追従性を評価した。

5.3. 定量的評価と他モデルとの比較

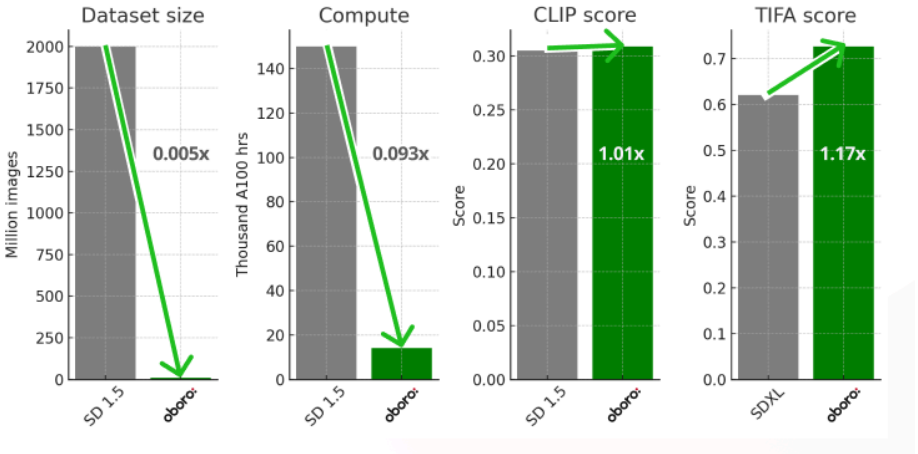
5.3.1. 各種評価指標

	SD 1.5	SDXL	oboro:
FID ↓	19.6	24.3	22.7
CLIP Score ↑	0.31	0.31	0.31
GenEval ↑	42%	54%	30%
TIFA Score ↑	0.52	0.62	0.85
Aesthetic Score ↑	4.8	6.4	4.5
Win Rate ↑	-	11.3%*	88.3%*

各指標の↑↓は↑が高いほうが良い指標、↓は低いほうが良い指標であることを表す。
*はoboro:animeとSDXLの比較結果

5.3.2. 学習時間・データセット

評価指標では競合モデルのStable Diffusion 1.5（SD1.5）またはStable Diffusion XL（SDXL）と遜色ない値を得た。学習時間・学習データセットともに少ない量の学習でも同等の性能を得ることができたことが大きな成果である。本プロジェクトでは著作権に配慮された少量のデータセットで基盤モデルの学習を行うことができるモデルを開発することが目標であったため、目標を達成できたと考える。既存モデルと比較して、200分の1のデータセット、10分の1の学習時間を達成している。



6. 結果と考察

6.1. 実験結果の総括

本研究では基盤モデルと追加学習モデルのふたつを開発した。独自の新規アーキテクチャを採用したことにより学習に必要な計算資源及びデータセットの削減に成功した。生成画像の品質は比較対象と遜色なく、同程度の品質を持つモデルをこれまでの10分の1の時間、50分の1の画像数で達成できた。著作権に配慮されたデータセットは画像数が少ないが、50分の1の画像数でも既存モデルと同程度の品質を得られたことは大きな優位性がある。基盤モデル「oboro:base」はアニメ事業者ごとに追加学習を行う前のモデルである。「oboro:anime」はアニメ事業者用に追加学習を行う想定で学習のテストを行ったモデルである。「oboro:anime」の学習においては、アニメ事業者からご提供いただく画像での学習を計画していたが、関係者間の合意形成や権利処理の枠組み整備をより丁寧に進める方針としたため、本検証フェーズではライセンス確認済みのイラストデータを用いた。また、権利者への配慮から「oboro:anime」は非公開としている。

6.2. モデルの特性と限界

本モデルの最大の長所は、oboroの命名の由来ともなった光と影、すなわち「陰陽」の表現力にある。明暗のコントラストを効果的に用いた、印象的な画像を生成することが可能である。色彩やコントラストの質も良好である。テキストエンコーダにT5_XXLを用いたことでプロンプトへの反応性も高く、意図した通りの画像を生成しやすい点も強みとして挙げられる。

一方で、いくつかの課題も確認されている。まず、ディテールを過剰に描画する傾向があり、その結果として画像全体にザラザラとした特有の質感が現れる場合がある。これは学習プロセスにおける過学習が原因である可能性がある。また、人物の描写は不得手な領域であり、安定した

品質での生成は今後の課題である。加えて、学習データセットの構成上、ファンタジーや特定のスタイルのイラストレーションといったジャンルの画像は生成できないという制約も存在する。

これらの不得意点は基盤モデルとしては大きな問題にならないとも考えられる。このプロジェクトでは基盤モデルに対し、アニメ事業者ごとに独自のデータセットを用いた追加学習を行うため、ファンタジーや人物キャラクターの学習を追加で行えるためだ。実際、oboro:animeの学習時にはキャラクター生成能力の向上が見られた。

6.3. 今後の課題と展望

本プロジェクトでは基盤モデルの開発と特化モデルの開発を行った。さらに、LoRAやControlNetなどの補助技術や活用ツール制作を含めた応用についても開発を進めている。また、本プロジェクトの知見を生かしてさらなるモデル開発を行う計画も行っている。

7. モデルの公開と利用方法

7.1. 公開リポジトリ（モデルデータ・推論コード）

https://huggingface.co/aihub-geniac/oboro

7.2. セットアップと実行方法

7.2.1. FLUX.1 [schnell]のライセンスに同意

本モデルではFLUX.1 [schnell]のVAEを利用して潜在空間から画像を復元します。画像エンコーダ/デコーダを利用するため、https://huggingface.co/black-forest-labs/FLUX.1-schnell のページを開き、ライセンスに同意してください。ファイルは画像生成実行時に自動的にダウンロードされるため、手動でのダウンロードは不要です。

7.2.2. aihub-geniac/oboroのファイルのダウンロード

https://huggingface.co/aihub-geniac/oboro に含まれるファイル全てをダウンロードしてください。

7.2.3. 依存関係のインストール

依存関係のライブラリを導入します。必要に応じて、適切な仮想環境にて実行してください。

```Shell
pip install -r requirements.txt
```

7.2.4. Hugging Face CLIへのログイン

Hugging Face CLI を用いて、必要モデルをダウンロードするため、事前にログインします。
コマンド実行後に、ご自身のHugging Faceアカウントのtoken（read権限のみでよい）の入力を求められますので、入力してください。

```Shell
huggingface-cli login
```

7.2.5. 画像生成の実行

```Shell
python src/infer.py --prompts 'A serene spring landscape outdoors,
featuring abundant cherry blossom trees with delicate pink flowers and
falling petals. A large body of reflective water, such as a lake or river,
is in the foreground, with a strong focus on the clear reflections of the
surrounding scene and sky on its surface. A picturesque wooden bridge with
```

```
a railing spans the water or crosses a path nearby. A path leads through
lush green grass and bushes along the riverbank or shore. A dense forest
covers rolling hills, and majestic mountains form a dramatic mountainous
horizon in the background under a cloudy sky. No humans are presented. best
quality, highly detailed.' --image_size '[416,736]' --cfg_scale '5.0'
--model_path 'oboro-base-v1-1b.safetensors' --output_dir 'output'
```

7.3. ライセンス

oboro:baseの学習済みモデルと推論用コードは、Apache License, Version 2.0にて公開しています。

8. 謝辞

本プロジェクトは、経済産業省（METI）および国立研究開発法人新エネルギー・産業技術総合開発機構（NEDO）が主導する日本の生成AI開発力強化プロジェクト「GENIAC」のもと、計算資源の援助を受けて実現されました。関係各位に深く御礼申し上げます。

9. 参考文献

1. High-Resolution Image Synthesis with Latent Diffusion Models, R. Rombach, A. Blattmann, D. Lorenz, P. Esser, B. Ommer, 2022年6月, CVPR'22, 10.1109/CVPR52688.2022.01042
2. Adding Conditional Control to Text-to-Image Diffusion Models,Lvmin Zhang, Anyi Rao, Maneesh Agrawala,2023年2月, https://arxiv.org/abs/2302.05543
3. LoRA: Low-Rank Adaptation of Large Language Models,Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, https://arxiv.org/html/2106.09685
4. Scalable Diffusion Models with Transformers,William Peebles, Saining Xie,2022年12月,https://arxiv.org/abs/2212.09748
5. Stable Diffusion 3 - Stability AI, 8月 5, 2025 https://stability.ai/news/stable-diffusion-3
6. https://laion.ai/blog/laion-5b/
7. Generative Adversarial Networks, Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, Yoshua Bengio, arXiv:1406.2661, https://arxiv.org/abs/1406.2661
8. Denoising Diffusion Probabilistic Models, Jonathan Ho, Ajay Jain, Pieter Abbeel, arXiv:2006.11239, https://arxiv.org/abs/2006.11239
9. Zero-Shot Text-to-Image Generation, Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, Ilya Sutskever, 2021年2月, arXiv:2102.12092, https://arxiv.org/abs/2102.12092
10. Learning Transferable Visual Models From Natural Language Supervision, Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, Ilya Sutskever,2021年2月, arXiv:2103.00020, https://arxiv.org/abs/2103.00020
11. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer, Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J. Liu, 2019年10月, arXiv:1910.10683, https://arxiv.org/abs/1910.10683
12. https://www.tensorflow.org/datasets/catalog/c4
13. https://deepmind.google/models/gemma/
14. Lumina-Image 2.0: A Unified and Efficient Image Generative Framework, Qi Qin, Le Zhuo, Yi Xin, Ruoyi Du, Zhen Li, Bin Fu, Yiting Lu, Jiakang Yuan, Xinyue Li, Dongyang Liu, Xiangyang Zhu, Manyuan Zhang, Will Beddow, Erwann Millon, Victor Perez, Wenhai Wang, Conghui He, Bo Zhang, Xiaohong Liu, Hongsheng Li, Yu Qiao, Chang Xu, Peng Gao, 2025年3月, arXiv:2503.21758, https://arxiv.org/abs/2503.21758
15. U-Net: Convolutional Networks for Biomedical Image Segmentation, Olaf Ronneberger, Philipp Fischer, Thomas Brox, 2015年5月, arXiv:1505.04597, https://arxiv.org/abs/1505.04597
16. https://huggingface.co/AuraDiffusion/16ch-vae
17. https://huggingface.co/openai/clip-vit-large-patch14
18. SDXL: Improving Latent Diffusion Models for High-Resolution Image Synthesis, Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, Robin Rombach, arXiv:2307.01952,

https://arxiv.org/abs/2307.01952
19. https://huggingface.co/laion/CLIP-ViT-bigG-14-laion2B-39B-b160k
20. FLUX.1 Kontext: Flow Matching for In-Context Image Generation and Editing in Latent Space, Black Forest Labs, Stephen Batifol, Andreas Blattmann, Frederic Boesel, Saksham Consul, Cyril Diagne, Tim Dockhorn, Jack English, Zion English, Patrick Esser, Sumith Kulal, Kyle Lacey, Yam Levi, Cheng Li, Dominik Lorenz, Jonas Müller, Dustin Podell, Robin Rombach, Harry Saini, Axel Sauer, Luke Smith, 2025年6月, arXiv:2506.15742, https://arxiv.org/abs/2506.15742
21. Lumina-Next: Making Lumina-T2X Stronger and Faster with Next-DiT, Le Zhuo, Ruoyi Du, Han Xiao, Yangguang Li, Dongyang Liu, Rongjie Huang, Wenze Liu, Lirui Zhao, Fu-Yun Wang, Zhanyu Ma, Xu Luo, Zehan Wang, Kaipeng Zhang, Xiangyang Zhu, Si Liu, Xiangyu Yue, Dingning Liu, Wanli Ouyang, Ziwei Liu, Yu Qiao, Hongsheng Li, Peng Gao, arXiv:2406.18583 https://arxiv.org/abs/2406.18583
22. GLU Variants Improve Transformer — Noam Shazeer, 2020年2月, arXiv:2002.05202, https://arxiv.org/abs/2002.05202
23. Rotary Position Embedding for Vision Transformer — Byeongho Heo, Song Park, Dongyoon Han, Sangdoo Yun, 2024年7月, arXiv:2403.13298, https://arxiv.org/abs/2403.13298
24. The Road Less Scheduled, Aaron Defazio, Xingyu Alice Yang, Harsh Mehta, Konstantin Mishchenko, Ahmed Khaled, Ashok Cutkosky, 2024年5月, arXiv:2405.15682, https://arxiv.org/abs/2405.15682
25. Matcha-TTS: A fast TTS architecture with conditional flow matching, Shivam Mehta, Ruibo Tu, Jonas Beskow, Éva Székely, Gustav Eje Henter, arXiv:2309.03199, https://arxiv.org/abs/2309.03199
26. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow, Xingchao Liu, Chengyue Gong, Qiang Liu, 2022年9月, arXiv:2209.03003, https://arxiv.org/abs/2209.03003
27. Improving and generalizing flow-based generative models with minibatch optimal transport, Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, Yoshua Bengio, 2023年2月, arXiv:2302.00482, https://arxiv.org/abs/2302.00482
28. https://huggingface.co/google/t5-v1_1-xxl
29. https://huggingface.co/datasets/madebyollin/megalith-10m
30. https://huggingface.co/Lin-Chen/ShareCaptioner
31. https://huggingface.co/microsoft/Florence-2-large
32. https://github.com/discus0434/aesthetic-predictor-v2-5
33. https://tech.preferred.jp/ja/blog/inference-for-data-preprocessing/
34. Real-Time Single Image and Video Super-Resolution Using an Efficient Sub-Pixel Convolutional Neural Network, Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, Zehan Wang,2016年9月, arXiv:1609.05158, https://arxiv.org/abs/1609.05158
35. Decoupled Weight Decay Regularization, Ilya Loshchilov, Frank Hutter, 2017年11月, arXiv:1711.05101, https://arxiv.org/abs/1711.05101
36. GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium, Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, Sepp Hochreiter, 2017年, Advances in Neural Information Processing Systems 30 (NIPS 2017), https://arxiv.org/abs/1706.08500
37. CLIPScore: A Reference-free Evaluation Metric for Image Captioning, Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, Yejin Choi,2021年4月,

arXiv:2104.08718,https://arxiv.org/abs/2104.08718
38. GenEval: An Object-Focused Framework for Evaluating Text-to-Image Alignment, Dhruba Ghosh, Hanna Hajishirzi, Ludwig Schmidt, 2023年10月, arXiv:2310.11513, https://arxiv.org/abs/2310.11513
39. TIFA: Accurate and Interpretable Text-to-Image Faithfulness Evaluation with Question Answering, Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, Noah A Smith, 2023年3月, arXiv:2303.11897, https://arxiv.org/abs/2303.11897
40. Image Aesthetic Assessment Based on Pairwise Comparison A Unified Approach to Score Regression, Binary Classification, and Personalization, Jun-Tae Lee; Chang-Su Kim, 2019 IEEE/CVF International Conference on Computer Vision (ICCV), Seoul, Korea (South), 2019, pp. 1191-1200, https://doi.org/10.1109/ICCV.2019.00128
41. Stable Diffusion 3: Research Paper — Stability AI, 2024年3月, https://stability.ai/news/stable-diffusion-3-research-paper
42. Introducing Stable Diffusion 3.5 — Stability AI, 2024年10月, https://stability.ai/news/introducing-stable-diffusion-3-5
43. RoFormer: Enhanced Transformer with Rotary Position Embedding — Jianlin Su, Yu Lu, Shengfeng Pan, Bo Wen, Yunfeng Liu, 2021年4月, arXiv:2104.09864, https://arxiv.org/abs/2104.09864
44. NormFormer: Improved Transformer Pretraining with Extra Normalization — Sam Shleifer, Jason Weston, Myle Ott, 2021年10月, arXiv:2110.09456, https://arxiv.org/abs/2110.09456
45. Sigmoid Loss for Language-Image Pre-Training (SigLIP) — Xiaohua Zhai, Alexander Kolesnikov, Basil Mustafa, Lucas Beyer, 2023年3月, arXiv:2303.15343, https://arxiv.org/abs/2303.15343
46. QK-Norm: Better Transformer Training with Query-Key Normalization — Chengnan Wang, Yunsheng Bai, Bohan Zhai, Cen Chen, 2023年, OpenReview（ICLR 2023）, https://openreview.net/forum?id=E4jFcMHgCQa
47. Rye — A Python Toolchain Manager — Astral, 2025年（アーカイブ/告知含む）, https://rye.astral.sh/
48. uv — An Extremely Fast Python Package Installer and Resolver — Astral, 2025年, https://docs.astral.sh/uv/
49. Cogview: Mastering text-to-image generation via transformers, Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al, Advances in Neural Information Processing Systems, 34:19822–19835, 2021.
50. 著作権法 第30条の4（情報解析のための複製等） — e-Gov法令検索（デジタル庁）, https://elaws.e-gov.go.jp/document?lawid=345AC0000000048
51. AIと著作権について（生成AIに関する考え方） — 文化庁, 2024年（令和6年）, https://www.bunka.go.jp/seisaku/bunkashingikai/chosakuken/houkokusho/94138001.html
52. Announcing Black Forest Labs, Black Forest Labs, 2024年8月, https://bfl.ai/blog/24-08-01-bfl