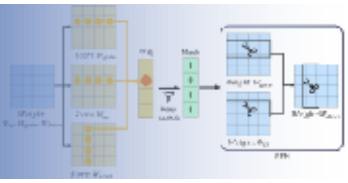
機械知覚&ロボティクスグループ
中部大学　工学研究科　ロボット理工学専攻

MPRG
MACHINE PERCEPTION AND ROBOTICS GROUP

2025年度　藤吉研究室　修士論文発表　アブストラクト

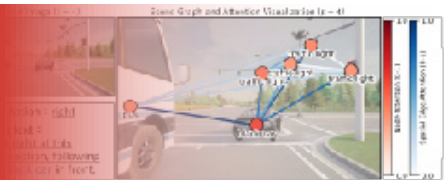
Pruning, LLM

AFRの構造化枝刈りへの適用におけるスコア集約手法の改良に関する研究
小林 亮太



Scene Graph, Natural Language Generation

時空間シーングラフを用いた案内文生成の高精度化と視覚的説明に関する研究
鈴木 颯斗

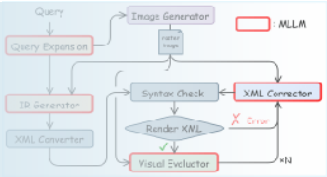


scRNA-seq, Contrastive learning

複数生物の遺伝子解析を行うMix-Geneformerに関する研究
西尾 優希

Multimodal Large Language Model, Vector Image Generation

MLLM による科学図の理解と生成に関する研究
増田 大河

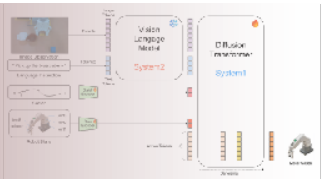


Reinforcement Learning, Large Language Model

深層強化学習における報酬関数の自動生成及び自動修正に関する研究
鈴木 佳三

Vision-language-action model, Robotics

スケッチ指示を用いた VLA モデルによるロボット動作性能の向上に関する研究
野田 修平



URL: http://mprg.jp

AFR の構造化枝刈りへの適用におけるスコア集約手法の改良に関する研究

TR24006 小林　亮太　　指導教授：藤吉 弘亘

1. はじめに

大規模言語モデル（LLM）は高い性能を示す一方，数十億のパラメータによる膨大な計算コストとメモリ使用量が実用化の障壁となっている．モデルサイズを軽量化するための代表的なアプローチとして冗長なパラメータ（重み）を削除する枝刈りがある．

枝刈りは非構造枝刈りと構造化枝刈りに大別される．非構造枝刈りは重み単位で独立に削除するため高い精度の維持を達成するが，削除位置が不規則なため推論の高速化には寄与しない．一方，構造化枝刈りはニューロン単位で重みを削除するため高速化が容易だが，削除の自由度が低いため性能劣化を招くという課題がある．

本研究では，非構造枝刈り手法である Adaptive Feature Retention (AFR) を構造化枝刈りに適用することで，重み単位の重要度（枝刈りスコア）の評価を活かして，モデルサイズの削減と推論の高速化を目指す．AFR を構造化枝刈りに適用する際には，重み単位の枝刈りスコアをニューロン単位に集約する必要がある．このとき，単純平均による集約では「符号情報の喪失」と「外れ値の影響」という 2 つの問題が生じる．本研究では，これらに対処するための改善手法を提案し，LLM として Llama-3-8B を用いた評価実験により提案手法の有効性を実証する．

2. Adaptive Feature Retention (AFR)

AFR [1] は，事前学習済みモデルに対する非構造枝刈り手法である．AFR は，ReFer [2] と SNIP [3] という 2 つの枝刈りスコアを標準化した後に加算することで，事前学習で獲得した特徴空間の維持と下流タスクへの適応を両立する．重み θ_n に対する枝刈りスコア $S_{AFR}(\theta_n)$ は式 (1) のように定義される．

$$S_{AFR}(\theta_n) = \mathcal{Z}\left(\left|\frac{\partial \sum_{l=1}^{L} F_{svd}^{l}}{\partial \theta_n}\theta_n\right|\right) + \mathcal{Z}\left(\left|\frac{\partial \mathcal{L}}{\partial \theta_n}\theta_n\right|\right) \quad (1)$$

ここで，$\mathcal{Z}(\cdot)$ は標準化を表し，第 1 項は ReFer，第 2 項は SNIP に相当する．F_{SVD}^{l} は，レイヤー l の出力する特徴量に対する特異値分解による特異値の平均である．ReFer は特徴表現の維持に焦点を当てているため，下流タスクへの学習の際に必要な重みを削除する可能性がある．

SNIP は，目的関数の勾配と重みの積を枝刈りスコアとする非構造枝刈り手法である．事前学習済みモデルでは多くの重みの勾配が小さく，$\frac{\partial \mathcal{L}}{\partial \theta_n} \approx 0$ となり，SNIP のみでは適切な枝刈りスコアの評価が困難である．

3. AFR の構造化における課題

本研究では，AFR の重み単位の枝刈りスコアの評価を構造化枝刈りに適用して，推論の高速化を行う．本章では，その際に生じる問題点を明確にする．

3.1 単純平均による構造化

非構造手法である AFR を構造化手法に適用するために図 1 に示すような単純平均による集約を考える．まず，重み行列の各要素に対して枝刈りスコアを算出し，式 (2) に示すようにニューロン単位で重みの枝刈りスコアを集約して平均スコア $\bar{S}_j$ を求める．

$$\bar{S}_j = \frac{1}{m}\sum_{i=1}^{m} |S_{ij}| \quad (2)$$

ここで，S_{ij} はニューロン j の i 番目の重みに対する枝刈りスコア，m はニューロンあたりの重み数である．最後に，ニューロン単位の平均スコアから低い順に枝刈り率に応じた数のニューロンを削除する．

3.2 単純平均による構造化の問題点

単純平均による枝刈りを LLM モデルに適用して，自然言語のデータセットで評価した場合，表 1 に示すように大幅な性能低下が生じた．この原因に関して，以下の 2 つの問題が存在すると考える．

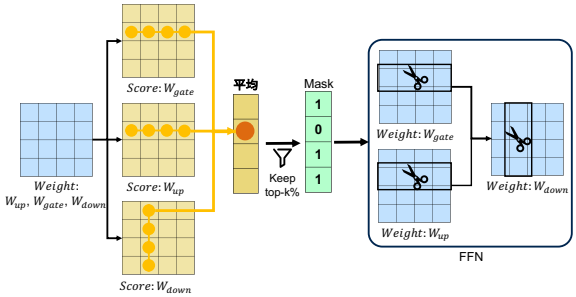


図 1: 構造化枝刈りのための枝刈りスコア集約の処理

表 1: 予備実験結果（accuracy %）

枝刈率	手法	WinoG	HellaS	ARC-e	ARC-c	MMLU	平均
0%	Llama-3-8B	72.61	79.16	77.74	53.33	62.13	68.99
20%	AFR（非構造）	71.59	73.79	76.94	48.38	58.85	65.91
	AFR（単純平均）	59.35	53.63	43.48	29.35	30.14	43.19
50%	AFR（非構造）	60.62	50.64	55.35	32.94	35.07	46.92
	AFR（単純平均）	52.25	29.29	29.92	26.02	23.07	32.11

問題 1: 最適化方向の一貫性情報の喪失

ReFer と SNIP の枝刈りスコアは勾配と重みの積として計算され，その符号は最適化過程における重みの挙動を表現する．勾配降下法による重みの更新則 $\theta \leftarrow \theta - \eta\frac{\partial \mathcal{L}}{\partial \theta}$ において，枝刈りスコアの符号は重みの最適化方向を示す指標となる．正の場合は重みの絶対値が減少する方向に，負の場合は増加する方向に働く．構造化枝刈りでは，ニューロン単位で削除を行うため，ニューロン内の重みの最適化方向の協調性が重要となる．全ての重みが同一方向に最適化される場合，そのニューロンは構造的に一貫した役割を持つ．一方，最適化方向が混在している場合，ニューロン内で相殺効果が生じており，全体としての寄与は限定的である．しかし，式 (2) では重み単位で絶対値を取るため，ニューロン内の最適化方向の一貫性という構造的特性が評価できず，ニューロン単位の重要度が適切に評価されない．

問題 2: 外れ値の影響

図 2 は，ある層における ReFer の重み単位の枝刈りスコア分布である．図 2 より，大部分の枝刈りスコアは比較的狭い範囲に分布しているが，両端に極端に大きな値や小さな値を持つ外れ値が少数存在していることがわかる．実際，全体の枝刈りスコアのレンジは，-2,884〜9,803 であり，その幅は 12,687 と大きな値である．ここで極値の 2%を削除するとレンジの幅は 48.799 となる．単純平均では，外れ値の有無によりニューロン間の平均スコアの大小関係が変化して，ニューロン単位の枝刈りスコアが適切に評価されないという問題がある．

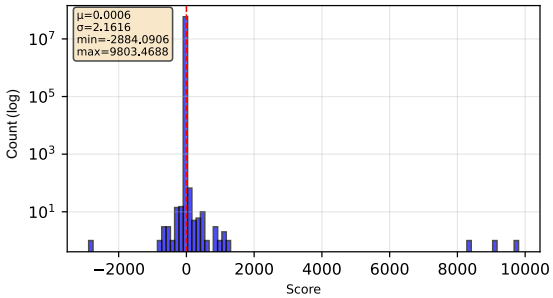


図 2: AFR の枝刈りスコアの分布例

4. 提案手法

前章で述べた 2 つの問題に対処するため，新たな枝刈りスコアの集約手法を提案する．

4.1 絶対値枝刈りスコア集約

問題 1 に対処するため，集約後の絶対値処理を提案する．ニューロンに含まれる各重みの枝刈りスコアを符号付きのまま平均し，平均化した後に絶対値を取る．これにより，符号が揃っているニューロンは平均後も大きな値を保ち，符号が混在しているニューロンは平均により小さくな

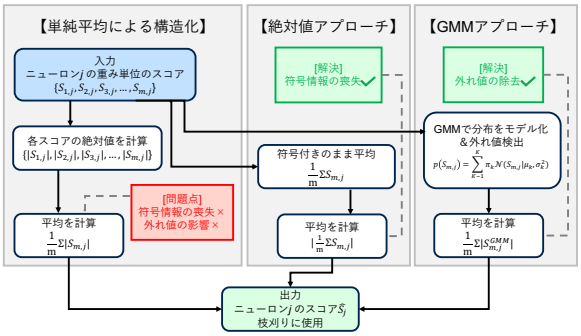


図 3: 提案手法の概要

る．このように，ニューロン内の符号の一貫性を評価でき，構造化枝刈りに適した枝刈りスコアの評価が可能となる．

4.2 GMM を用いた外れ値処理

問題 2 の外れ値の影響を軽減するため，Gaussian Mixture Model（GMM）を用いた外れ値処理手法を提案する．ニューロン j の枝刈りスコア集合 $\{S_{1j}, S_{2j}, \ldots, S_{mj}\}$ に対し，GMM により枝刈りスコア分布をモデル化する．各枝刈りスコア S_{ij} の確率密度 $p(S_{ij})$ を以下のように定義する．

$$p(S_{ij}) = \sum_{k=1}^{K} \pi_k \mathcal{N}(S_{ij}|\mu_k, \sigma_k^2) \tag{3}$$

ここで，π_k, μ_k, σ_k^2 はそれぞれ混合比，平均，分散であり，EM アルゴリズムで推定する．成分数 K は BIC により $K \in \{1,2,3,4,5\}$ から最適な値を選択する．推定されたモデルにより各枝刈りスコアの密度を評価し，密度が下位 2%に該当するものを低密度と判定する．次に，枝刈りスコアを昇順にソートし，分布の両端から連続して低密度が続く範囲を外れ値として検出し，その境界の枝刈りスコアで置換する．これにより，極端な値の影響を抑制しつつ，データ数を保ったまま安定した集約が可能となる．

4.3 提案手法の枝刈りスコア

4.1 節と 4.2 節で提案する手法を組み合わせた手法も提案する．これは，符号付きの重み単位の枝刈りスコアに対して GMM 処理を適用した後，平均を取り，最後に絶対値を取る処理となる．具体的には，GMM 処理後の枝刈りスコア S'_{ij} を用いて以下のように計算する．

$$\bar{S}_j^{abs+GMM} = \left| \frac{1}{m} \sum_{i=1}^{m} S'_{ij} \right| \tag{4}$$

この手法により，符号の一貫性評価と外れ値の影響抑制の両方を実現し，安定した枝刈りスコアの評価が期待される．

5. 評価実験

提案手法の有効性を検証するため，Llama-3-8B を対象とした評価実験を実施した．

5.1 実験概要

Llama-3-8B の各ブロックにおける FFN に対して構造化枝刈りを適用する．枝刈り率は既存研究で広く採用されている 20% と 50% とする．性能評価には，WinoGrande, HellaSwag, ARC-easy/ARC-challenge, MMLU の 5 つのベンチマークデータセットを使用し，accuracy を評価指標とする．比較手法として，非構造 AFR，構造化 AFR（単純平均，絶対値のみ，GMM のみ，絶対値+GMM），および既存手法（LLM-Pruner，LoRAP，CFSP）を用いる．

5.2 実験結果

表 2 に提案手法の評価結果を示す．単純平均と比較して提案手法である絶対値のみの場合は 20% 枝刈りで 13.04 ポイント，GMM のみの場合は 16.31 ポイントの精度向上を示した．絶対値+GMM の場合が最も高い性能を示し，20% 枝刈りで 18.06 ポイント，50% 枝刈りで 11.33 ポイントの精度向上を達成した．これは，符号情報の損失と外れ値の影響の両方が解決されたことによる相乗効果である．

5.3 既存手法との比較

提案手法（絶対値+GMM）を既存の構造化枝刈り手法と比較する．表 3 に既存手法との比較結果を示す．提案手法は 20% 枝刈りで 61.25% を達成し，他の既存手法を上回った．50% 枝刈りでも 43.44% を達成し，LLM-Pruner（34.87%）と比較して 8.57 ポイント向上した．

表 2: 提案手法の精度評価（accuracy %）

枝刈率	手法	WinoG	HellaS	ARC-e	ARC-c	MMLU	平均
0%	Llama-3-8B	72.61	79.16	77.74	53.33	62.13	68.99
20%	非構造枝刈り						
	AFR	71.59	73.79	76.94	48.38	58.85	65.91
	構造化枝刈り						
	AFR（単純平均）	59.35	53.63	43.48	29.35	30.14	43.19
	AFR（絶対値のみ）	67.40	67.90	70.20	42.83	32.81	56.23
	AFR（GMM のみ）	68.57	**70.70**	70.58	44.80	42.86	59.50
	AFR（絶対値+GMM）	**68.90**	69.70	**72.81**	**46.16**	**49.32**	**61.25**
50%	非構造枝刈り						
	AFR	60.62	50.64	55.35	32.94	35.07	46.92
	構造化枝刈り						
	AFR（単純平均）	52.25	29.29	29.92	26.02	23.07	32.11
	AFR（絶対値のみ）	52.96	37.88	43.18	25.94	23.04	36.60
	AFR（GMM のみ）	**58.41**	**48.59**	48.36	29.78	27.78	42.58
	AFR（絶対値+GMM）	56.75	47.33	**51.89**	**29.95**	**28.27**	**43.44**

表 3: 既存手法との比較（accuracy %）

枝刈率	手法	WinoG	HellaS	ARC-e	ARC-c	MMLU	平均
0%	Llama-3-8B	72.61	79.16	77.74	53.33	62.13	68.99
20%	提案手法	68.90	69.70	**72.81**	**46.16**	49.32	**61.25**
	LLM-Pruner	69.85	69.02	63.59	40.53	48.36	58.26
	LoRAP	**71.19**	**70.48**	69.61	45.48	44.57	60.27
	CFSP	68.67	68.06	67.63	42.92	**50.71**	59.60
50%	提案手法	56.75	**47.33**	**51.89**	**29.95**	**28.27**	**43.44**
	LLM-Pruner	52.49	35.50	37.88	25.51	22.95	34.87
	LoRAP	**57.30**	40.19	42.60	26.79	26.85	38.75
	CFSP	57.06	43.13	48.53	28.50	26.61	40.77

5.4 リソース削減効果の評価

表 4 に各枝刈り率におけるリソース削減効果を示す．50% 枝刈りでパラメータ数及び VRAM 使用量 35.1% 削減，推論速度 1.57 倍の高速化を達成した．構造化枝刈りにより実用的な高速化が実現された．

表 4: リソース削減効果の評価

枝刈率	パラメータ数		VRAM [GB]		推論速度 [ms/sample]	
0%	8.03B	—	16.06	—	22.98	1.0x
20%	6.90B	14.0%↓	13.08	14.0%↓	20.89	1.10x
50%	5.21B	35.1%↓	10.42	35.1%↓	14.64	1.57x

6. おわりに

本研究では，非構造枝刈り手法を構造化枝刈りに適用することにより，精度の維持と推論の高速化を両立する手法を提案した．

単純平均による枝刈りスコアの評価では，符号情報の損失と外れ値の影響により性能が著しく劣化することを発見した．これらに対処するため，絶対値アプローチと GMM による外れ値除去を組み合わせた手法を提案した．Llama-3-8B を対象とした評価実験により，提案手法は単純平均手法と比較して大幅な性能向上を達成し，既存の構造化枝刈り手法と比較して同等以上の性能を示した．また，50% 枝刈りでパラメータ数及び VRAM 使用量 35.1% 削減，推論速度 1.57 倍の高速化を達成した．

今後の課題として，Attention モジュールへの拡張，集約方法の最適化，層ごとの適応的枝刈り率設定が挙げられる．

参考文献

[1] 新田常顧 等, “事前学習済みモデルの知識維持と下流タスク適応を両立した Single-shot Foresight Pruning”, 画像の認識・理解シンポジウム, 2025

[2] 新田常顧 等, “事前学習モデルの特徴表現を維持した Single-shot Foresight Pruning”, 画像の認識・理解シンポジウム, 2024

[3] N. Lee, *et al.*, “SNIP: Single-shot Network Pruning based on Connection Sensitivity”, International Conference on Learning Representation (ICLR) , 2019.

研究業績

[1] 小林亮太，平川翼，山下隆義，藤吉弘亘, “非構造的枝刈り手法の構造化手法への変換による LLM の軽量化”, 東海支部連合大会, 2025.

時空間シーングラフを用いた案内文生成の高精度化と視覚的説明に関する研究

TR24009 鈴木颯斗　　指導教授：藤吉 弘亘

1. はじめに

運転者の認知的負担を抑えつつ，走行状況を直感的に理解できる案内を提供することが，快適な運転ナビゲーション支援では重要である．このようなナビゲーションシステムの実現においては，車両が運転手に対して運転シーンに合わせて適切な情報を提示する必要がある．既存システムは，地図情報に基づいた経路案内や定型的な音声案内が主流であり，複雑かつ動的に変化する走行環境では，運転手が直感的に状況を把握することは容易ではない．このような課題に対し，周囲の状況を踏まえて人間のように案内を行う Human-like Guidance に関する研究が注目されている．

本研究では，Human-like Guidance の実現に向け，視界情報を基にした環境認識と自然言語生成を統合し，運転手が直感的に理解可能な案内文を生成することを目標とする．走行シーンにおける判断対象となる情報は多岐にわたり，特に時間的変化を伴う環境認識では，情報量の増加が生成精度や安定性に影響を及ぼす可能性がある．そこで本研究では，車両の視界情報から得られるオブジェクトの空間的・時間的関係を時空間シーングラフとして表現し，これを基に案内文を生成する手法を提案する．さらに，生成したシーングラフに対し，Graph Attention Networks(GAT)[1] を用いて案内に重要な対象を強調しながら情報を統合することを目指す．そして，推論時に得られる Attention を可視化することで，生成された案内文に対する視覚的説明を可能とする．

2. 関連研究

本研究では，車両の視界情報を基に環境を理解し，運転状況に即した案内文を生成することを目的としている．本章では，この目的に関連する技術を述べる．

2.1 動画像からのキャプション生成

キャプション生成は，画像または動画像を入力とし，その内容を自然言語による文章として生成するタスクである．本タスクでは，一般に視覚特徴を抽出するエンコーダと，抽出された特徴に基づいて文章を生成するデコーダからなる Encoder-Decoder 構成が採用される．視覚特徴抽出の手法として，画像を対象とする場合には CNN，動画像を対象とする場合には 3DCNN や時系列情報を考慮可能な Transformer ベースのモデルが広く用いられる．文章生成には，Transformer に代表される自己回帰型の言語モデルが用いられ，Cross-Attention 機構を通じて視覚特徴と単語列を統合しながら逐次的に自然言語の説明を生成する．これにより，入力画像や動画像の内容と意味的に整合性のあるキャプション生成が可能となる．

2.2 シーングラフによる環境理解

Graph Neural Network を視覚認識へ応用した手法として，画像中のオブジェクトをノードとして表現し，その関係性をグラフ構造として扱うシーングラフに基づく手法が提案されている．Graph R-CNN[2] は，物体検出モデルによって得られたオブジェクト間の関係をシーングラフとして明示的に構築することで，画像の構造的理解が向上することを示している．

3. 提案手法

本研究では，走行シーンの動画像から得られるオブジェクトの時空間的関係を時空間シーングラフとして構築し，GAT を用いた Graph-to-Text モデルにより案内文を生成する手法を提案する．また，Graph Encoder における Attention を可視化することで，案内文生成の判断根拠を視覚的に提示し，モデルの解釈性と信頼性の向上を図る．提案手法の全体構成を図 1 に示す．

3.1 マルチオブジェクトトラッキング

交通シーンには多様なオブジェクトが存在し，その外観も大きく変化する．従来のシーングラフ構築では，オブジェクトをノードとして定義した場合に，この多様な外観情報を含めることが困難である．そこで本研究では，Open-Vocabulary 物体検出モデルである YOLO-World[3] を用いる．YOLO-World は画像特徴とテキスト特徴を統合したクロスモーダル表現により，未学習クラスの zero-shot 検出が可能である．本手法では，テキストで指定したクラスラベルを直接ノードのラベルとして利用することで，ノード表現を簡潔かつ解釈可能な形式に統一する．

また，オブジェクト自身の時間的な差分をシーングラフとして表現する場合，検出オブジェクトをフレーム間で一貫して追跡する必要がある．本手法では追跡アルゴリズムである BoT-SORT[4] を用い，各オブジェクトに一貫した ID を付与する．これにより，シーングラフ構築時に時間方向の解析が可能となる．

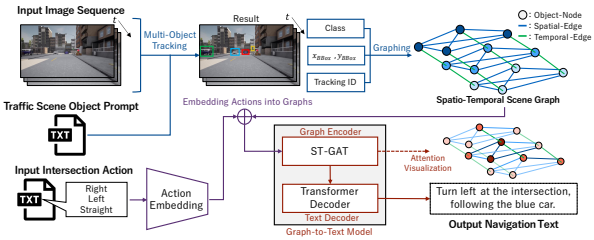


図 1: 提案手法のアーキテクチャ

3.2 時空間シーングラフの構築

動画像が与えられたとき，前述したマルチオブジェクトを行い，検出オブジェクトの位置・クラス情報・追跡情報を用いて時空間シーングラフ G を式 (1) として構築する．

$$G = \{\boldsymbol{V}, \boldsymbol{E}\} \tag{1}$$

ここで，$\boldsymbol{V}$ はノード集合，$\boldsymbol{E}$ はエッジ集合である．

ノード集合 $\boldsymbol{V}$ の定義： 各フレーム t におけるノード集合 $\boldsymbol{V}_t$ を，マルチオブジェクトトラッキングによって得られた検出オブジェクトの集合として，式 (2) のように定義する．

$$\boldsymbol{V}_t = \{v_i^t \mid (B_i^t, c_i^t, id_i^t) \in \mathcal{D}_t\} \tag{2}$$

ここで，$\mathcal{D}_t$ はマルチオブジェクトトラッキングの出力を表し，B はオブジェクト毎の境界ボックス座標，c はクラスラベル，id は ID 化されたトラッキング情報を示す．

エッジ集合 $\boldsymbol{E}$ の定義： 同一フレーム内のオブジェクト間の関係性を空間的エッジ $\boldsymbol{E}_{\text{spatial}}$ として式 (3) のように定義する．

$$\boldsymbol{E}_{\text{spatial}} = \{((v_i^t, v_j^t), w_{ij}^t)\}, w_{ij}^t = \| \tilde{B}_i^t - \tilde{B}_j^t \|_2 \tag{3}$$

ここで，w_{ij}^t はエッジに対する重みを示し，オブジェクト間のユークリッド距離を付与する．これにより，フレーム毎のオブジェクト間の動的変化をグラフ内に表現する．

続いて，時系列方向におけるオブジェクト間の関係性として時系列エッジ $\boldsymbol{E}_{\text{temporal}}$ を式 (4) のように定義する．

$$\boldsymbol{E}_{\text{temporal}} = \{((v_i^t, v_j^{t+1})) \mid id_i^t = id_j^{t+1}\} \tag{4}$$

この処理では，前後のフレームでトラッキング ID が一致するノードに対してエッジが接続される．

最終的に，エッジ集合 $\boldsymbol{E}$ は式 (5) となる．

$$\boldsymbol{E} = \boldsymbol{E}_{\text{spatial}} \cup \boldsymbol{E}_{\text{temporal}} \tag{5}$$

3.3 シーングラフへの Action 埋め込み操作

案内文生成では，シーンの状況だけでなく，自車の行動 Action（右折・直進など）を考慮することが重要である．本手法では，ナビゲーション時に与えられる Action をテキスト形式として入力し，埋め込み層を通して Action 特徴を得る．そして，得られた Action 特徴を前段で生成されたシーングラフ内のすべてのノード特徴へ統合する．事前にシーングラフに Action を埋め込むことで，Action に基づいて着目すべきノードが強調されるような効果を図る．

表 1: 各モデルで生成された案内文の精度結果

Method	5 frame				10 frame				15 frame			
	B-1	B-4	M	R	B-1	B-4	M	R	B-1	B-4	M	R
3DCNN	0.568	0.322	0.575	0.643	0.551	0.292	0.538	0.617	0.519	0.268	0.515	0.601
3DResNet	0.459	0.197	0.446	0.559	0.448	0.173	0.439	0.547	0.457	0.180	0.449	0.534
VTN	0.583	0.337	0.578	0.565	0.412	0.142	0.378	0.537	0.379	0.099	0.377	0.471
ViViT	0.524	0.266	0.538	0.592	0.540	0.285	0.551	0.603	0.549	0.274	0.559	0.611
Ours	**0.610**	**0.363**	**0.635**	**0.668**	**0.617**	**0.382**	**0.646**	**0.675**	**0.631**	**0.388**	**0.649**	**0.677**

※ B-1：BLEU-1, B-4：BLEU-4, M：METEOR, R：ROUGE

3.4 Graph-to-Text モデル

本研究では，時空間シーングラフから文章の生成を行う Graph-to-Text モデルを提案する．Graph-to-Text モデルは，グラフの特徴抽出を行う Graph Encoder と文章生成を行う Text Decoder で構成される．Graph Encoder では，空間方向と時系列方向に分けて Attention を適用し，特徴抽出を行う Spatial Temporal GAT (ST-GAT) を構築する．Text Decoder では，Graph Encoder により抽出されたグラフ特徴量から，Transformer Decoder を用いて文章の生成を行う．また，モデルの推論時，最終層における各エッジに対する Attention スコアをグラフ上に可視化することで，モデルの判断根拠の解釈を可能とする．

4. データセット

ナビゲーションタスクには，走行車両の車載カメラ映像と対応する案内文のペアからなるデータセットが必要となる．本研究では案内文生成に特化したデータセットを CARLA Simulator を用いて独自に作成する．撮影には 8 つのマップを用い，撮影条件は以下の通り設定する．

- フレームレート：10 fps
- 天候条件：ClearNoon, WetNoon
- 撮影範囲：交差点約 50m 手前から交差点通過直後

案内文のアノテーションは手動で実施し，注目対象に基づいた案内文を作成する．作成したデータセットは，合計 160 シーン，計 10,219 フレームで構成される．各シーンには，前述した案内文，進行方向における動作情報が含まれる．

5. 評価実験

評価実験を通じて提案手法の有効性を検証する．本実験では，ベースライン手法との比較，入力フレーム数が 5 フレーム，10 フレーム，15 フレームにおける異なるフレーム長が案内文の生成精度に与える影響について分析する．評価には，BLEU，METEOR，ROUGE を用いる．

5.1 ベースライン手法

ベースライン手法として動画像から直接特徴量を抽出する手法を用いる．具体的には，提案手法におけるシーングラフを構築する過程と Graph Encoder を Video Encoder に置き換える．本実験では，CNN および Transformer をベースとした，3DCNN，3DResNet，Video Transformer Network (VTN)，Video Vision Transformer (ViViT) を用いる．

5.2 実験条件

学習設定は，学習率 1.0×1.0^{-4}，エポック数 100，バッチサイズ 32，Dropout 率 0.3 とする．学習の最適化アルゴリズムには AdamW を用いる．これらの設定は，提案手法およびベースライン手法の全てのモデルで統一する．

5.3 定量的評価

提案手法およびベースライン手法の各モデルで生成された案内文の精度について定量的評価によって比較を行う．評価結果を表 1 に示す．結果より，提案手法は全てのフレーム数において他の手法を上回る精度を達成しており，フレーム数が増加するほどより顕著に精度が向上していることが確認できる．

5.4 定性的評価

提案手法およびベースライン手法の各モデルで生成された案内文について定性的に評価する．各手法における案内文生成結果の例を図 2 に示す．結果より，Groud Truth と同様の “yellow car” を中心とした案内文を生成できているものは提案手法のみであり，最も適切な説明となっている．ベースライン手法においては，最も動作の変化が大きい “black car” もしくは画像内に存在しないオブジェクトを注目しており，不適切な説明となっている．

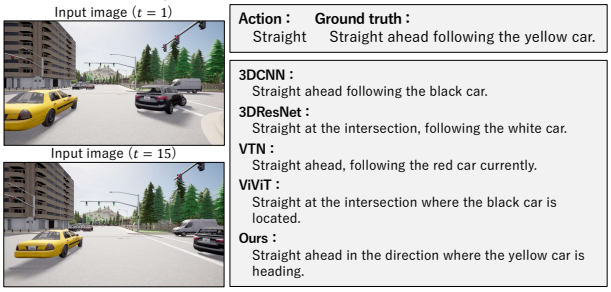


図 2: 各手法における案内文生成結果

次に，提案手法における案内文生成において，推論時の Attention を時空間シーングラフ上に可視化する．Attention の可視化結果を図 3 に示す．結果より，グラフ上では “black car” に着目しており，生成案内文の着目しているオブジェクトと一致する．また，エッジはオブジェクト間の関連度として解釈することができる．したがって，モデルが生成した案内文の判断根拠をグラフを通して視覚的に説明可能であることを示している．

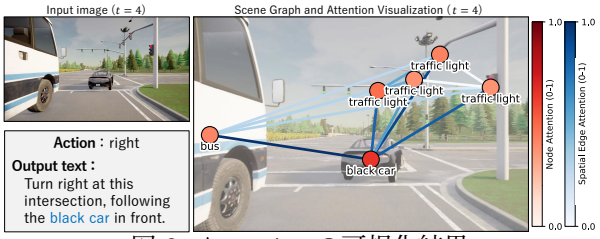


図 3: Attention の可視化結果

6. おわりに

本研究では，車両の視界情報から動的な環境を理解し，運転手に直感的な案内文を生成する手法を提案した．走行シーンのオブジェクト関係を時空間シーングラフとして表現し，Graph-to-Text モデルにより案内文生成を行った．また，GAT による重要情報の強調と Attention 可視化による判断根拠の提示を実現した．評価実験では，提案手法が CNN や Transformer ベースの Video Encoder を用いたベースライン手法よりも高い精度を示し，特に長期間の情報統合において有効性が確認された．さらに，定性的評価および Attention 可視化から，モデルが適切な対象に注目し案内文を生成していることを確認した．

今後の課題として，より複雑な環境や多様な運転シナリオへの適用とその有効性の検証が挙げられる．

参考文献

[1] V.Peter, *et al.*, “Graph Attention Networks”, ICLR, 2018.

[2] Y.Jianwei, *et al.*, “Graph R-CNN for Scene Graph Generation”, ECCV, 2018.

[3] T.Cheng, *et al.*, “YOLO-World: Real-Time Open-Vocabulary Object Detection”, CVPR, 2024.

[4] N.Aharon, *et al.*, “BoT-SORT: Robust Associations Multi-Pedestrian Tracking”, arXiv, 2022.

研究業績

[1] H. Suzuki, *et al.*,“Enhancing Navigation Text Generation and Visual Explanation Using Spatio-Temporal Scene Graphs with Graph Attention Networks”, ITSC, 2025.

(他 学会発表 3 件)

複数生物の遺伝子解析を行う Mix-Geneformer に関する研究

TR24010 西尾優希　　指導教授：藤吉 弘亘

1. はじめに

次世代シーケンサにより単一細胞の遺伝子発現量を計測できるようになり，細胞ごとの特性解析が可能となっている．解析の属人化を避け，効率化するために，深層学習を用いた single-cell RNA sequencing (scRNA-seq) 解析手法として，Geneformer[1] や Mouse-Geneformer[2] が提案されている．これらは，それぞれヒトとマウスの遺伝子発現情報を文章として扱い，Transformer により学習することで，汎用的な細胞の特徴表現を獲得している．この学習により獲得した特徴表現を用いることで，細胞型分類や in silico 摂動などの下流タスクで高い性能を示している．Geneformer および Mouse-Geneformer は，いずれも単一生物種を対象とした事前学習モデルであり，生物種を横断した解析は困難である．一方，生物種を横断した解析が可能となれば，マウスで得られた解析結果をヒトの解析に適用でき，創薬プロセスの短縮や研究効率の向上が期待できる．そこで本研究では，ヒトおよびマウスの scRNA-seq データを統合的に学習する Mix-Geneformer を提案する．これにより，生物種を横断した解析が可能なモデルの構築を目指す．

2. 深層学習を用いた scRNA-seq 解析

scRNA-seq 解析は，次世代シーケンサにより細胞を単一細胞レベルに分離して計測した遺伝子発現量をもとに，細胞間の多様性や状態変化を解析する手法である．scRNA-seq 解析における課題として，前処理や特徴量設計，細胞型同定などの解析工程において解析者の判断が介在する場面が多く，属人的なバイアスが生じやすい．この課題に対し，解析の自動化と汎用的な特徴表現の獲得を目的とした手法が提案されている．

深層学習を用いて細胞の特徴表現を学習する手法として，Geneformer[1] および Mouse-Geneformer[2] が提案されている．これらは Transformer を用いた scRNA-seq 解析手法であり，細胞を文章，遺伝子をトークンとして扱う点に特徴がある．具体的には，各細胞において遺伝子発現量上位 2,048 個の遺伝子を抽出し，発現量順に整列したトークン列として細胞文を構成する．両モデルはいずれも Masked Language Modeling（MLM）による事前学習を通じて，細胞型分類や in silico 摂動などの下流タスクに応用可能な特徴表現を獲得している．一方で，これらのモデルは単一生物種のデータを用いて事前学習しているため，生物種を横断した統合解析には至っていない．

3. 提案手法：Mix-Geneformer

本研究では，生物種を横断した解析が可能なモデルの実現を目的として，ヒトおよびマウスの scRNA-seq データを同一の Transformer で学習する scRNA-seq 解析モデル Mix-Geneformer を提案する．ヒトおよびマウスの scRNA-seq データを統合して学習することで，生物種に依存しない細胞表現の獲得を目指す．

3.1 Mix-Geneformer における事前学習

Mix-Geneformer の事前学習では，Masked Language Modeling（MLM）と SimCSE を組み合わせて用いる．MLM は，各細胞文内における遺伝子の関係を学習するための自己教師あり学習である．SimCSE は，ミニバッチ内の細胞文同士の関係性を捉えるための対照学習である．Mix-Geneformer の学習方法の概要を図 1 に，損失関数を式 (1)〜(3) に示す．

$$L_{\text{total}} = L_{\text{MLM}} + L_{\text{SimCSE}} \tag{1}$$

$$L_{\text{MLM}} = -\sum_{i \in M} \log P(x_i \mid x_{\setminus M}; \theta) \tag{2}$$

$$L_{\text{SimCSE}} = -\frac{1}{N}\sum_{i=1}^{N} \log \frac{\exp\left(\text{sim}(h_i^{(1)}, h_i^{(2)})/\tau\right)}{\sum_{j=1}^{N} \exp\left(\text{sim}(h_i^{(1)}, h_j^{(2)})/\tau\right)} \tag{3}$$

式 (2)，(3) において，M は一部をマスクしたトークンの集合，$\text{sim}(\cdot,\cdot)$ はコサイン類似度，τ は温度パラメータである．式 (2) の MLM 損失 (L_{MLM}) は，マスクしたトークンを予測する過程を通じて，細胞文内における遺伝子発現順位の関係性を学習する．式 (3) の SimCSE 損失 (L_{SimCSE}) は，同一の細胞文に対してエンコーダ内の確率的な dropout により得られる 2 つの特徴表現を生成し，それらを正例対として扱う．各細胞 i に対して，同一エンコーダを 2 回通すことで得られる表現 $h_i^{(1)}$ および $h_i^{(2)}$ を正例とし，同一ミニバッチ内の他の細胞に由来する表現 $\{h_j^{(2)}\}_{j \neq i}$ を負例として対照学習を行うことで，細胞間の特徴表現の類似度を学習する．

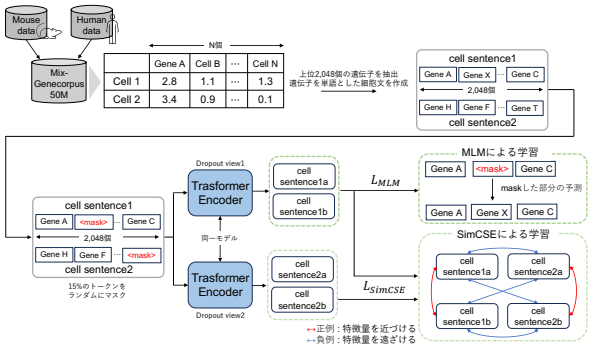


図 1: Mix-Geneformer の学習方法

3.2 学習データセット:Mix-Genecorpus-50M

Mix-Geneformer の学習データセットとして，ヒトの scRNA-seq データセットである Genecorpus-30M とマウスの scRNA-seq データセットである Mouse-Genecorpus-20M を統合し，約 5,000 万細胞から構成する Mix-Genecorpus-50M を作成した．Genecorpus-30M および Mouse-Genecorpus-20M は，複数の公開データセットを統合しており，多様な臓器や細胞型を含んでいる．

3.3 事前学習

本研究では，ヒトおよびマウスの scRNA-seq データを同一の事前学習データとして扱うことで，生物種を横断した解析が可能なモデルの作成を目的とする．事前学習では，埋め込み特徴を 256 次元とし，6 層の Transformer エンコーダを用い，MLM および SimCSE に基づく対照学習を組み合わせて学習を行った．MLM では，細胞文中の一部のトークンをランダムにマスクし，周辺のトークンから元のトークンを予測することで，細胞文内における遺伝子間の関係性の学習を促した．SimCSE による対照学習では，特徴表現間の類似度に基づく損失を導入することで，各細胞文同士の類似性を学習させた．事前学習は，バッチサイズ 8，warmup10,000 ステップを含む 10 エポックで行った．

4. 評価実験

本研究では，Mix-Geneformer の評価として，細胞型分類と in silico 摂動実験の 2 種類の下流タスクを行う．各実験において，単一生物種内の評価と，生物種の横断性の評価を行う．いずれの下流タスクにおいても，事前学習済みの Transformer に対して 10 エポックのファインチューニングを行う．

4.1 細胞型分類による評価

細胞型分類タスクでは，単一生物種内の評価として，事前学習済みモデルに対してマウスおよびヒトのデータでファインチューニングし，Geneformer および Mouse-Geneformer と細胞型分類精度を比較する．さらに，生物種の横断性を評価するため，マウスの脾臓データでファインチューニングしたモデルをヒトの脾臓データに，ヒトの脾臓データでファインチューニングしたモデルをマウスの脾臓データに適用し，細胞型ごとのモデルの特徴表現を UMAP により

可視化する．表 1 および表 2 に，マウスおよびヒトデータにおける分類精度を示す．

表 1: マウスの細胞型分類精度

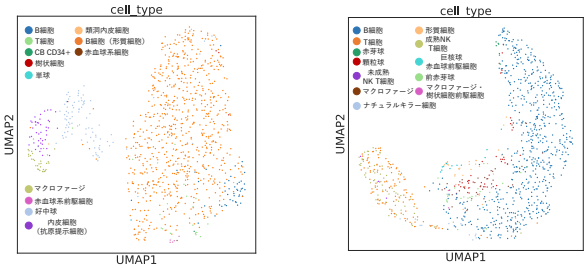
Organ	Types	Mouse-GF	Mix-GF
Brain	15	96.9	**97.6**
Heart	11	**97.8**	97.7
Kidney	18	94.9	**95.4**
Large_intestine	7	93.1	**94.6**
Limb muscle	9	99.5	**99.7**
Mammary gland	7	99.0	**99.1**
Spleen	10	**98.7**	98.6
Thymus	6	97.0	**97.6**
Tongue	3	94.9	**95.3**

表 2: ヒトの細胞型分類精度

Organ	Types	Human-GF	Mix-GF
Spleen	6	98.9	**99.0**
Brain	6	96.8	**97.7**
Immune	10	94.4	**95.1**
Kidney	15	92.8	**93.3**
Large_intestine	16	92.7	**93.4**
Liver	12	91.1	**91.2**
Lung	16	93.4	**94.3**
Pancreas	15	93.0	**93.5**
Placenta	3	97.9	**98.2**

表 1 および表 2 より，マウスおよびヒトのいずれのデータにおいても，Mix-Geneformer は従来モデルと同等以上の分類精度を示した．この結果は，複数の生物種のデータを同時に学習に用いることで，細胞型分類に寄与する特徴を獲得した可能性を示唆している．

また，生物種の横断性を評価した UMAP 可視化結果を図 2 に示す．図 2 より，可視化対象と異なるデータでファインチューニングしたモデルであっても，UMAP 上で細胞型ごとに一定程度クラスタが分離していることを確認した．このことから，Mix-Geneformer はマウスとヒトのデータを同時に学習することで，生物種を横断した解析が可能であると示唆される．



(a) マウスで FT →ヒトの脾臓データ可視化　(b) ヒトで FT →マウスの脾臓データ可視化

図 2: 生物種の横断性に関する UMAP 可視化結果

4.2 in silico 摂動実験による評価

in silico 摂動実験とは，コンピュータ上で遺伝子の過剰発現や遺伝子削除を模擬し，細胞状態を目標状態へ近づける上で重要な遺伝子を同定する手法である．in silico 摂動実験の概要を図 3 に示す．

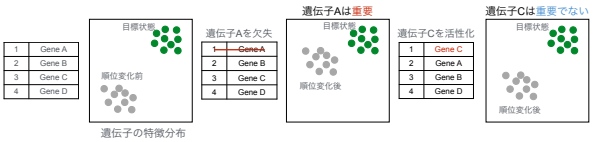


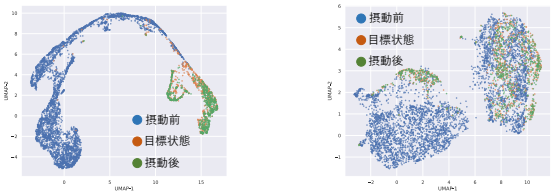
図 3: in silico 摂動実験の概要

本実験では，対象遺伝子の順位を変化させることで細胞に仮想的な摂動を与え，摂動後の特徴表現と目標状態の特徴表現間の類似度を算出する．その上で，目標状態への変化が大きい遺伝子を重要遺伝子として特定する手順で実験を行う．評価指標には，cosine_shift（↑）および p_value（↓）を用いた．cosine_shift はコサイン類似度の変化量であり，正の値が大きいほど目標状態へ近づくことを意味する．また，p_value は統計的有意性を評価する指標であり，本研究では $p < 0.05$ を統計的に有意とする．

本研究では，単一生物種内の評価として，マウスの心臓疾患データでファインチューニングしたモデルを用い，マウスにおける心臓疾患状態から正常状態へ変化させる in silico 摂動実験を行う．また，生物種間の横断性の評価として，マウスの心臓疾患データでファインチューニングしたモデルをヒトの心臓疾患データに適用し，心臓疾患状態から正常状態へ変化させる in silico 摂動実験を行う．前者の実験は遺伝子の削除，後者の実験は遺伝子の過剰発現による実験を行っている．これらの実験において，Mix-Geneformer が重要と判定し，実際の生物実験で有効性が確認された遺伝子の一部を表 3 に，摂動前，摂動後，および目標状態における細胞の特徴表現を UMAP により可視化した結果を図 4 に示す．

表 3: in silico 摂動実験で確認された有効遺伝子の一部

使用モデル	遺伝子名	cosine_shift	p_value
マウス	ALDOB	0.011	1.56E-2
マウス	ALDH3B2	0.011	9.03E-3
ヒト	MTRNR2L11	0.202	0.0
ヒト	NAP1L6	0.035	1.66E-03



(a) ヒトで FT したモデルによる実験　(b) マウスで FT したモデルによる実験

図 4: in silico 摂動実験における UMAP 可視化

表 3，図 4 から，マウスおよびヒトデータでファインチューニングしたモデルの両者ともに，摂動後の細胞の特徴表現は摂動前と比較して目標状態に近づいた．cosine_shift の摂動前後の変化は，図 4(a) においては約 0.43，図 4(b) においては約 0.05 であり，定量的，定性的評価ともに in silico 摂動実験の成功を確認した．一方で，ヒトデータでファインチューニングしたモデルでは，UMAP によるクラスタ間の分離がより明瞭であり，摂動前後における細胞の特徴表現の変化量も大きいことを確認した．これは，マウスデータで学習したモデルをヒトデータに適用する際に，生物種の違いに起因するドメインギャップが存在する可能性を示唆している．この差異を解消するには，マウスとヒト間でのデータ正規化や，種間の関係性学習のための損失を定義する必要があると考える．

5. おわりに

本研究では，ヒトおよびマウスの scRNA-seq データを同一の Transformer で事前学習するモデル Mix-Geneformer を提案した．細胞型分類タスクにおいて，Mix-Geneformer は従来モデルと同等以上の精度を示し，同一種内における性能の有効性を確認した．また，生物種の横断性の評価として，UMAP による特徴表現の可視化を行った結果，異なる生物種でファインチューニングしたモデルであっても，一定程度のクラスタ構造が保持されることを示した．in silico 摂動実験においても同様の傾向を確認し，マウスデータでファインチューニングしたモデルでもヒトの in silico 摂動実験が可能である．一方で，同一種のデータでファインチューニングを行った場合と比較すると性能に差が見られることから，生物種の差異が結果に影響する可能性が示唆される．以上より，Mix-Geneformer は生物種を横断した scRNA-seq 解析が可能である一方で，種間差異をより適切に扱うための学習手法の設計が今後の課題である．具体的には，生物種の差異に起因する影響の緩和，ヒトおよびマウス以外の生物種への拡張が挙げられる．

参考文献

[1] C. V. Theodoris, *et al.*, “Transfer learning enables predictions in network biology,” *Nature*, 2023.

[2] K. Ito, *et al.*, “Mouse-Geneformer: A deep learning model for mouse single-cell transcriptome and its cross-species utility,” *PLOS Genetics*, 2025.

研究業績

[1] 西尾優希，山下隆義，伊藤啓太，平川翼，藤吉弘亘，”Mix-Geneformer: Unified Representation Learning for Human and Mouse scRNA-seq Data”，IIBMP，2025

MLLM による科学図の理解と生成に関する研究

TR24014　増田 大河　　指導教授：藤吉 弘亘

1. はじめに

学術論文における科学図は，複雑な概念や構造，関係性を直感的に理解させるための重要な役割を担っている．科学図を自動生成することで，研究者の図作成プロセスの支援が可能である．学術論文中の科学図の多くがベクタ形式で表現されるため，ベクタ形式に基づく科学図の自動的生成が求められている．AutomaTikZ[1] は，大規模言語モデル (LLM) を活用して LaTeX の TikZ パッケージのコード生成を行うことでベクタ形式の科学図生成を実現している．しかし，AutomaTikZ には以下の 2 つの問題がある．(i) 構文エラーなどを含むコードが生成されることがある．(ii) 生成結果を人間が修正するには TikZ パッケージの専門知識が必要となる．

本研究では，専門的な記述言語に対する知識を必要とせず，作図ツール上で人間が直感的に編集可能な形式である XML を対象とする．そして，MLLM を用いた科学図の自動生成手法を提案する．具体的には，エラーの自動修正と生成結果の自己改善を行う機能を導入する．これにより，構造的な整合性を保ちつつ，高品質なベクタ形式の科学図を生成可能となる．また，生成された XML 形式の科学図は既存の作図ツール上で容易に修正・拡張・再利用ができる．そのため，図の作成から改良に至る反復的な作業プロセスの効率化に貢献する．

2. AutomaTikZ

Belouadi らは，科学図を対象として TikZ コードの自動生成手法である AutomaTikZ を提案している [1]．AutomaTikZ は，CLIP による画像特徴を事前学習済みの LLaMA に統合したモデルをファインチューニングし，自然言語キャプションと真値となる図の画像から TikZ コードを生成する．これにより，テキストと図の整合性を考慮した高品質な TikZ コードの生成を可能にしている．

生成性能の評価には DaTikZ データセットが用いられている．DaTikZ は，インターネット上から収集された 約 12 万件もの TikZ コードと自然言語キャプションのペアから構成される大規模データセットである．データは，TeX Stack Exchange の投稿，arXiv 論文の TeX ソース，および教育目的の TikZ 図共有サイトなど，実用的に利用されている公開リソースから収集されている．AutomaTikZ は，DaTikZ データセットを用いて学習することで TikZ コードの自動生成を実現している．しかし，生成対象が TikZ に限定されている点や，構文エラーを含むコードを生成する場合があるといった課題がある．

3. 提案手法: XML-Diagram Agent

ベクタ形式の科学図の記述方法には，TikZ や SVG など多様な形式が存在し，それぞれ異なる目的に基づいて設計されている．AutomaTikZ で対象とされている TikZ は TeX 用の描画パッケージであり高度な数理表現が可能である．しかし，文法が複雑であり，人間による図の追加修正等は TikZ の専門知識が必要となる．一方で，XML はノード・エッジ・レイアウト情報が明確に分離された構造として記述でき，draw.io のような作図ツール上で追加修正可能である．

本研究では，クエリから XML の科学図を生成するフレームワークである XML-Diagram Agent (XDA) を提案する．XDA は，ラスタ画像生成モデルと MLLM を用いてエラーの修正や図の品質の改善を自律的に行うことで，高品質なベクタ形式の科学図を生成する．XDA のフレームワークを図 1 に示す．本フレームワークは以下のモジュールを組み合わせて構築する．

Query Expansion

Query Expansion は与えられたクエリを基に，図を構成する要素や構造をプロンプト文に変換する．これにより，曖昧さの少ない構造的な情報を後続のモジュールに入力で

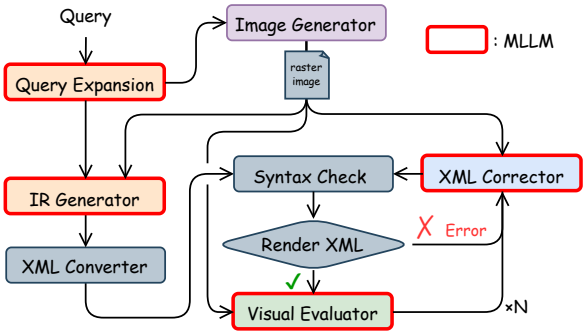


図 1: XDA のフレームワーク

き，図の意味的整合性および生成品質の向上が期待できる．

Image Generator

Image Generator (IG) では，プロンプト文を画像生成モデルに入力してラスタ形式の画像を生成する．ここで生成した画像は，後続のモジュールに入力してデザイン的な補助情報として活用する．

IR Generator

IR (Intermediate Representation) Generator では，プロンプト文とラスタ画像から図の構成要素やそれらの関係をグラフ表現として生成する．これにより，ノードやエッジといった要素の種類，接続関係，階層構造などを明示的に表現することができる．また，IR Generator で生成したグラフ表現は，後続の XML Converter で XML のドラフトをルールベースの作成に利用する．

Visual Evaluator

Visual Evaluator では生成した XML をレンダリングした図が，クエリとラスタ画像に従っているかを視覚的に評価し，不整合や改善点を抽出する．評価結果は自然言語によるフィードバックとして出力され，後続の XML Corrector に提供される．

XML Corrector

XML Corrector では XML とラスタ画像と Visual Evaluator が出力したフィードバック文を入力として受け取り，改善した XML を生成する．生成した XML は外部ツールによって構文チェックおよび再レンダリングを行い，その結果を再び Visual Evaluator に入力する．この処理を反復することで，図の構造的および視覚的な品質を段階的に向上させる．

ここで，ラスタ形式の画像を生成する Image Generator 以外のモジュールに MLLM を利用する．最終的にフレームワークは，以上 4 つのモジュールに加えてルールベースで動作する XML Corrector と外部ツールによる Sytax Check，Render XML を組み合わせて，段階的かつ自律的にエラーを改善し，図の品質を向上させるフィードバックループによって構成される．

4. 評価実験

提案手法の有効性を検証するために，科学図の生成性能の比較を行う．

4.1. 実験概要

評価実験を行うために，科学図の XML データをインターネット上から収集して DiagramXML データセットを構築した．DiagramXML は，インターネット上から収集した 科学図の XML と，説明文から構成される．説明文は，XML をレンダリングして得られた画像に対して GPT-4o を用いて生成した．説明文の正確さと図の完成度を人間による 0 から 100 のスコアリングにより検証し，70 以上のスコアであった 70 件を利用する．

表 1: 各手法の評価結果

生成手法	w/ IG	CLIPScore	C-BLEU	DiagramEval						SR
				Node			Path			
				prec.	recall	F1	prec.	recall	F1	
Zero-Shot XML	-	71.38	5.760	0.601	0.480	0.518	0.252	0.175	0.174	0.69
Zero-Shot graph	-	85.61	**6.332**	0.859	0.708	0.752	0.396	0.290	0.300	**0.99**
XDA		84.79	6.060	0.857	0.725	0.762	0.440	0.332	0.336	0.97
	✓	**87.32**	6.123	**0.872**	**0.783**	**0.802**	**0.516**	**0.443**	**0.426**	**0.99**

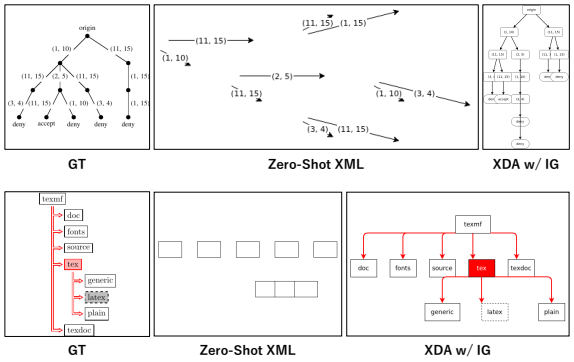


図 2: 各手法により生成された科学図

4. 2. 定量的評価

定量的評価では，提案手法である XDA と，2 種類の Zero-Shot 手法との比較を行う．具体的には，Zero-Shot prompting により XML を直接生成する手法 (Zero-Shot XML) と，中間表現としてグラフ表現を生成した後に XML に変換する手法 (Zero-Shot graph) を用いる．これらの手法においては，いずれも MLLM として Qwen2.5-VL-72B-Instruct モデルを使用する．また，XDA における IG で利用する画像生成モデルは，gpt-image-1 モデルとする．さらに，XDA における IG の有効性を確認するために IG の有無による比較も行う．評価指標として CLIPScore，C-BLEU，DiagramEval[2]，生成成功率 (SR) を用いる．

各手法の評価結果を表 1 に示す．これより，CLIPScore と DiagramEval の評価値に注目すると，提案手法である IG あり XDA が最高精度を達成していることが確認できる．これより，ラスタ画像生成モデルの性能も活かしつつ自己改善を行うことで高品質なベクタ形式の科学図を生成できていると言える．また，Zero-Shot XML は全ての評価指標で Zero-Shot graph を超える精度を達成しており，科学図生成においてグラフ表現を中間表現として用いることが有効であることがわかった．

さらに，XDA におけるラスタ画像生成モデルの有無による精度に着目すると IG あり XDA は IG なし XDA と比較して全ての評価指標で高い精度を示している．これより，ラスタ画像生成モデルで生成した画像を基にフィードバックと修正を繰り返して生成した科学図は，より高品質な科学図を生成できているといえる．

以上より，提案手法である IG あり XDA は，Zero-Shot prompting 手法とは異なり，自己改善ループによる安定した構造生成を可能とし，成功率および構造的な一致率の観点で優れた性能を示しており，高品質かつ構造的に整合性の取れた XML 形式の科学図の自動生成が可能であることがわかった．

4. 3. 定性的評価

提案手法である IG あり XDA により生成した場合と Zero-Shot prompting 手法で XML を直接生成した場合を定性的に比較する．各手法により生成された科学図を図 2 に示す．これより，IG あり XDA は Zero-Shot prompting 手法で XML を直接生成した場合と比べて，より正解画像に近い図を生成できていることがわかる．特に矢印の関係性が大きく向上しており，IG あり XDA 手法の有効性を確認した．

さらに，IG あり XDA 手法における Visual Evaluator による生成図の変化を図 3 に示す．これより，Visual Evaluator で生成されたフィードバック文を用いて修正された科学図は回数を重ねるごとに真値に近づいていることがわかる．特に，図中のオブジェクトの配置や色味などが改善されており，生成結果によるフィードバック文の指摘を踏まえて改善できたといえる．

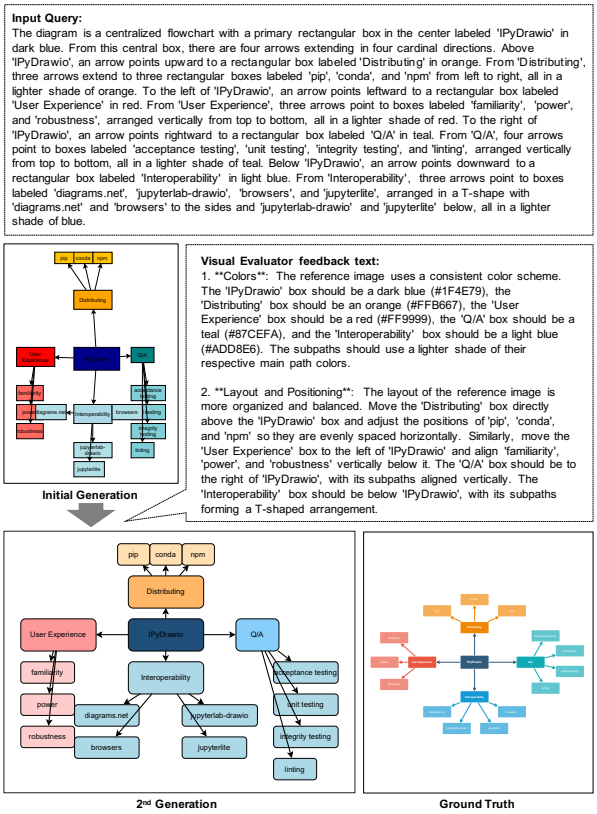


図 3: Visual Evaluator による生成図の変化

5. おわりに

本研究では，グラフ構造を中間表現として利用し，生成結果に対するフィードバックを活用した自己改善を行うフレームワークである XDA を提案した．実験結果より，提案手法は直接 XML を生成する場合と比較して，生成の安定性および構造的正確性の観点で優れた性能を示すことを確認した．また，特にグラフ表現を中間表現として用いることが極めて有効であることを確認した．今後の課題としては，より多様な科学分野における図表への適用や，レイアウトや視認性といった視覚的品質のさらなる向上が挙げられる．

参考文献

[1] Jonas Belouadi, *et al.*, “AutomaTikZ: Text-Guided Synthesis of Scientific Vector Graphics with TikZ” CILR. 2024.

[2] Chumeng Liang, *et al.*, “Evaluating LLM-Generated Diagrams as Graphs” EMNLP. 2025.

研究業績

[1] 増田 大河 等,“TikZAgent: LLMs による科学ベクタ図の自動生成”, 画像の認識・理解シンポジウム (MIRU), 2025. （他 3 件）

深層強化学習における報酬関数の自動生成及び自動修正に関する研究

TR24008 鈴木 佳三　　指導教授：藤吉 弘亘

1. はじめに

強化学習 (RL) は，エージェントが環境との相互作用を通じて方策を学習する機械学習手法の一種であり，ロボット制御やゲーム攻略などの分野で応用が進んでいる．RL では，環境から与えられる報酬をもとに行動を評価し，報酬を最大化するように方策を更新する．そのため，報酬関数の設計はエージェントの学習や性能を左右する重要な要素である．一方で複雑なタスクでは報酬の設計が難しく，専門知識や試行錯誤への依存が大きな課題となっている．この問題に対し，大規模言語モデル (LLM) を用いて報酬関数を自動生成・修正する Text2Reward (T2R) [1] が提案されている．これにより，自動で報酬関数を生成できる一方で，生成された報酬関数が必ずしも実行可能であるとは限らず，環境の仕様と不整合なコードや未定義変数を含む場合がある．また，報酬関数の修正において人間による評価やフィードバックを前提としており，設計者の主観や負担に依存する点が課題として残されている．

本研究では，これらの課題に対処するため，報酬関数の自動生成および自動修正を安定して実現するフレームワークを提案する．提案手法では，LLM が生成した報酬関数の実行可能性を担保する Auto Debug Module と，自動的に RL 結果を分析する Feedback LLM を導入することで，人間に依存しない報酬関数の生成および改善を目指す．

2. RL における報酬設計と従来法

RL において，報酬関数はエージェントの行動を数値的に評価し，学習の方向性を決定づける重要な要素である．エージェントは報酬を最大化するように方策を更新するため，報酬設計は最終的な方策の性質や学習性能に大きな影響を与える．

2.1 人手による報酬設計

一般的な RL では，タスクの目的を人間が解釈し，目標状態への到達や制約条件の遵守などを評価基準として，状態や行動に応じた報酬を定義する．報酬は単一の項目で与えられる場合もあるが，多くの場合は複数の評価項目を組み合わせた加算形式で表現される．この際，各評価項目に対する重み付けや，疎報酬か密報酬かといった報酬形状を人手で設定する．これらの設定は探索の容易さや学習の安定性に強く影響するため，学習曲線やエージェントの行動例を観察しながら報酬関数を反復的に修正する試行錯誤が必要となる．専門家はタスク構造や RL アルゴリズムの特性を踏まえた調整が可能である一方，非専門家にとっては，どの評価項目をどの程度強調すべきかを判断することが難しい．その結果，報酬関数の品質に差が生じ，意図しない行動の誘発や学習の停滞が生じることがある．

2.2 従来手法：Text2Reward

T2R は，LLM を用いて報酬関数を自動生成し，人間のフィードバックを通じて修正する手法である．図 1 に示すように，まず LLM に対して RL タスクの概要や環境情報を与え，報酬関数のコードを生成させる．生成された報酬関数を用いて RL を実施し，その学習結果やエージェントの振る舞いをもとに，人間がフィードバックを与えることで報酬関数を更新する．

T2R は，従来人手に依存していた報酬設計を自動化する可能性を示している．一方で，実際の RL タスクに適用する際には，生成された報酬関数の実行可能性や，人間によるフィードバックの与え方に起因する問題がある．そこで，T2R を用いた事前調査を行い，これらの問題点を明らかにする．

2.3 事前調査 1：生成関数の実行可能性

本調査では，LLM が生成する報酬関数の実行可能性について検証する．LLM に対して，ManiSkill2 の PushChair-v1 を対象とした報酬関数の生成を依頼し，生成された関数を実際の RL 環境上で実行する．実行時にエラーが発生した場合は，当該関数を実行不可能と判定し，新たに報酬関数の生成を行う．この操作を繰り返し，10 個の実行可能な報酬関数が得られるまで試行する．

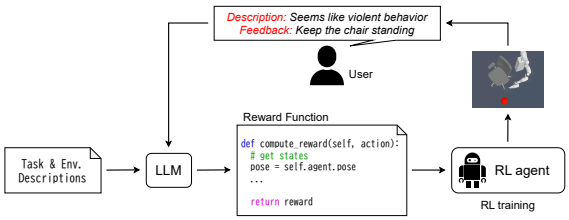


図 1: Text2Reward

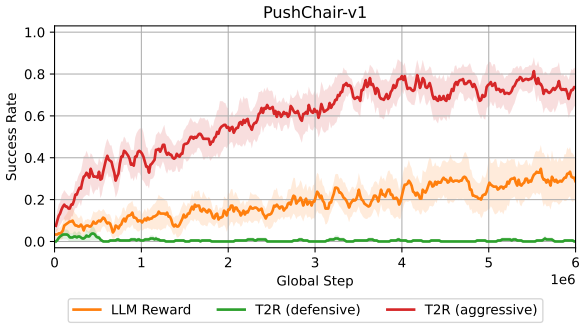


図 2: フィードバック方針の違いによる RL への影響

結果を表 1 に示す．この調査により，LLM が生成した報酬関数には，未定義の変数や環境に存在しない属性を参照する記述が含まれる場合が多く，実行可能な関数を得るためには複数回の生成が必要であることがわかった．このことから，T2R では報酬関数生成の段階で実行可能性が十分に担保されておらず，LLM による生成に不安定性が存在することが確認できる．

表 1: T2R における報酬関数の実行可能性

LLM	総生成回数	実行可能率 [%]
GPT-4o	36	27.8
GPT-5	38	26.3

2.4 事前調査 2：修正方針の違いによる RL への影響

本調査では，報酬関数の修正時に用いるフィードバックの違いが与える RL への影響について調査する．GPT-5 が生成した報酬関数 (LLM Reward) を異なる 2 つの方針で修正し，それぞれで RL を実施する．一つは安全性や必要最低限の動作を重視する保守的な修正方針 (defensive) であり，もう一方はタスクの達成速度や効率を重視する積極的な修正方針 (aggressive) である．これらのフィードバックに基づいて報酬関数を修正し，同一条件下で RL を実行した．RL タスクには，PushChair-v1 を用いる．

結果を図 2 に示す．aggressive は最高で 80%程度のタスク成功率を示したが，defensive は終始 0%付近で停滞している．この調査により，フィードバック方針の違いによって学習の進行や最終的な性能に大きな差が生じることが確認された．同じ初期報酬関数を用いた場合であっても，人間の判断による修正内容の違いがタスク成功率やエージェントの行動に大きく影響する．この結果は，T2R における報酬修正が人間の熟練度や価値観に強く依存していることを示しており，非専門家にとって安定した性能向上を達成することが困難であるという課題を示唆している．

3. 提案手法：Auto-Text2Reward

本研究では，図 3 に示すように，報酬関数の自動生成および自動修正を安定して実現するためのフレームワークを提案する．本手法では，Code Generate LLM，Auto

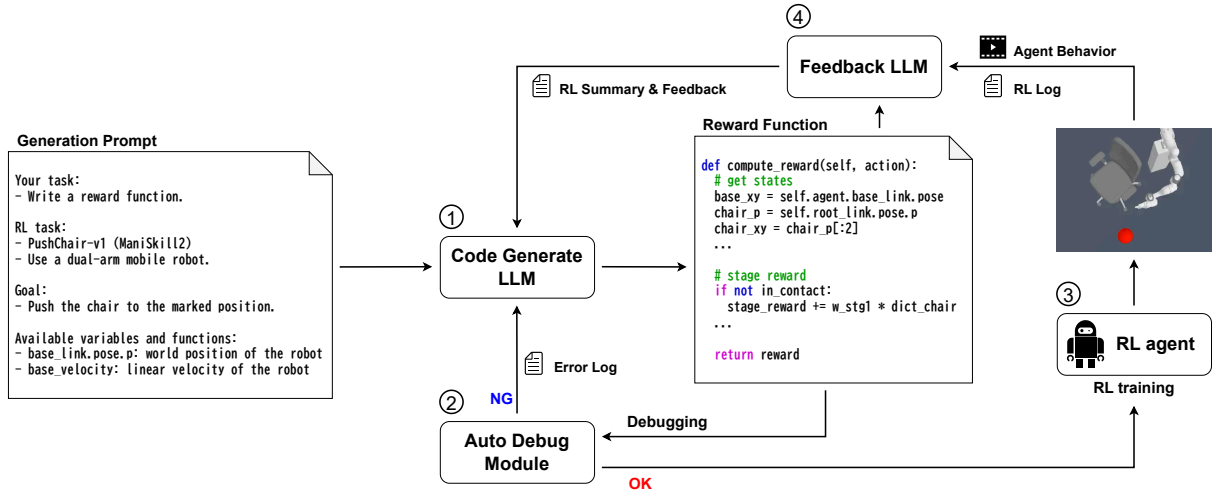


図 3: 提案手法 (Auto-Text2Reward)

Debug Module，Feedback LLM を組み合わせ，T2R における生成段階および修正段階の不安定性の低減を図る．

3.1 報酬関数の自動生成

存在しない変数の参照や記述ミスを抑制するために，以下の手順に従い報酬関数を生成する．まず，①の Code Generate LLM が，RL 環境で直接使用可能な変数や関数を明示したプロンプトを基に，報酬関数を生成する．次に，②の Auto Debug Module に生成した報酬関数を渡し，環境において正常に実行可能であるか検証する．ここでエラーが発生した場合，Auto Debug Module はエラーログを LLM に共有し，再生成を行うことで実行可能な報酬関数のみを選別する．正常に実行可能な報酬関数が得られたら，③の実際の RL 環境に渡り，RL を実施する．

3.2 報酬関数の自動修正

人間の熟練度や判断基準に依存しないように，RL の結果を自動的に分析する Feedback LLM を導入する．RL の実施後，④の Feedback LLM を用いて RL の分析を行う．ここでは，学習中のタスク成功率や獲得報酬の推移などの RL ログデータ，エージェントの振る舞いを記録した画像列を入力として受け取り，学習状況やエージェントの振る舞いを分析する．その後，使用した報酬関数と分析結果を踏まえ，報酬関数の改善案であるフィードバックを生成する．これらの分析結果とフィードバックに基づき，①の Code Generate LLM が再び報酬関数を生成する．

4. 評価実験

提案手法の有効性を検証するため，報酬関数の実行可能性と，報酬設計の品質に関する評価実験を行う．

4.1 報酬関数の実行可能性

生成した報酬関数の実行可能性を検証するため，ManiSkill2 のタスクを対象として報酬関数を生成する．事前調査 1 と同様に，LLM には GPT-5 を用いて，10 個の実行可能な報酬関数を生成するまでに要した試行回数を計測する．

結果を表 2 に示す．提案手法は，いずれのタスクにおいても従来手法より少ない試行回数で実行可能な報酬関数を生成しており，プロンプト改善および Auto Debug Module が報酬関数の実行可能性を向上させていることが確認できる．

表 2: 提案手法における報酬関数の実行可能性

タスク	手法	総生成回数	実行可能率 [%]
PushChair-v1	T2R	38	26.3
	Ours	14	**71.4**
OpenCabinetDoor-v1	T2R	72	13.9
	Ours	11	**90.9**
OpenCabinetDrawer-v1	T2R	75	13.3
	Ours	14	**71.4**

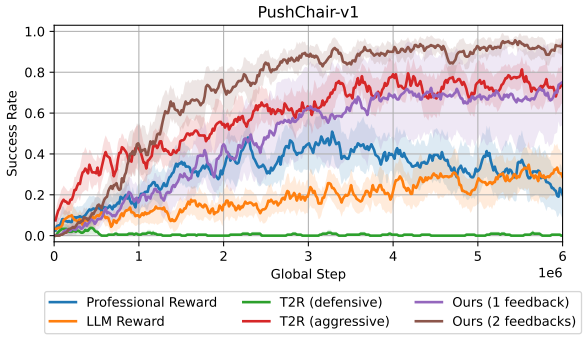


図 4: タスク成功率の推移

4.2 報酬設計の品質

生成した報酬関数の品質を検証するため，PushChair-v1 を用いて RL を実施する．比較対象として，専門家設計の Professional Reward，GPT-5 が生成した LLM Reward，T2R によって LLM Reward を保守的および積極的な方針で修正した T2R (defensive/aggressive)，提案手法によって n 回の自動修正を行った Ours (n feedback) を用いる．

各エージェントの学習過程におけるタスク成功率を図 4 に示す．提案手法は，いずれの報酬関数よりも高いタスク成功率を達成した．また，提案手法による修正を繰り返すことで，修正前の報酬関数を用いた学習より性能が向上している．以上より，提案手法は人間を介せず，報酬関数の自動生成および自動修正を実現したといえる．

5. おわりに

本研究では，RL における報酬設計の困難さに着目し，報酬関数の自動生成および自動修正を安定して実現するフレームワークを提案した．T2R の課題に対し，提案手法では，生成された報酬関数の実行可能性を担保する Auto Debug Module と，RL の結果を分析してフィードバックを生成する Feedback LLM を導入した．評価実験より，提案手法は専門家が設計した報酬関数や従来手法により得られた報酬関数と比較して，エージェントの性能を改善させることが可能であることを確認した．

今後は，RL 分析サマリーのさらなる最適化や，動画情報の導入方法，フィードバック生成頻度の制御，あるいは軽量な言語モデルとの併用といった観点から，計算効率と分析精度の両立を検討する．

参考文献

[1] T. Xie, *et al.*, “Text2Reward: Reward Shaping with Language Models for Reinforcement Learning”, ICLR, 2024.

研究実績

[1] 鈴木佳三 等, ”MaskDP による事前学習のマルチドメイン拡張”, 日本ロボット学会学術講演会, 2024.

スケッチ指示を用いた VLA モデルによるロボット動作性能の向上に関する研究

TR24012 野田修平　　指導教授：藤吉 弘亘

1. はじめに

言語指示と視覚観測に基づいてロボットの行動を直接生成する Vision-Language-Action（VLA）モデルが注目されている [1]．VLA は，事前に獲得した世界知識を利用することで未知のタスクにおいても汎化可能である．しかし，軌道や回り込み方向，速度変化，停止位置などの動作指示に関する詳細を言語のみで正確に表現することは困難である．

このような言語指示の曖昧さを補うため，視覚的に意図を与えるスケッチ指示を用いる手法が提案されている．RT-Sketch[2] は，手描きのスケッチ指示を目標表現として用いることで，言語目標が曖昧な場合や視覚的外乱が存在する場合でも，空間的な意図を伝達できる可能性を示した．

そこで，本研究では VLA モデルに対してスケッチ指示を導入する．従来の言語指示に加えて，スケッチ指示を用いることで，言語指示による高い汎用性・認識能力を活かしつつ，動作の具体的な意図を補完し，より人の意図をくみ取ったロボット動作の実現と動作性能の向上を目指す．

2. Vision-Language-Action（VLA）モデル

VLA モデルは，視覚情報と言語指示から環境・タスクを理解し，ロボットの状態（関節角など）を条件として，関節角やグリッパなどの行動を直接出力する．これにより End-to-End な制御を実現する．また，汎用的な理解能力と高速な動作生成を両立するため，高レベルの解釈・推論と運動制御を分担させる 2 層構造（dual-system）の VLA も提案されている．

GR00T N1[3] は，視覚・言語モデルと拡散モデルからなる 2 層構造の VLA である． GR00T N1 の構造を図 1 に示す．視覚・言語モデル部分 (System2) では，環境・指示内容を解釈し，観測画像と言語指示をトークン列として入力することで，視覚言語特徴を抽出する．拡散モデル (System1) 部分では，ロボットの各関節角度などの状態情報を実機の構成に合わせた MLP で埋め込み，拡散過程で用いるノイズ付与済み行動と拡散時刻を Action Encoder で埋め込む．これらと視覚・言語モデルで得られた特徴量の cross-attention を求めることで，環境や言語指示を考慮した行動系列を生成する．System1 は 16 ステップ先までの行動を生成し，高頻度に更新することで滑らかな実機制御を可能にする．

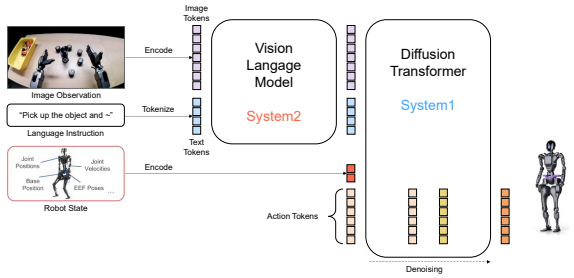


図 1: GR00T N1 のモデル構造

3. 提案手法

本研究では，GR00T N1 をベースとし，Diffusion Transformer にスケッチ指示を入力することで，意図した軌道・速度でのロボット動作を実現する．

3.1 スケッチ入力に対応した VLA モデル

モデル構造を図 2 に示す．視覚言語特徴とロボットの状態ベクトルに加えて，スケッチ指示を動作生成の条件情報として Diffusion Transformer へ入力する．これにより，従来の言語指示のみでは指定が難しい回り込み方向や通過経路などを条件情報として動作に直接反映する．

3.2 スケッチ指示

スケッチ指示は画像上の座標 (x, y) を記録する．その後，変化量（$\Delta x, \Delta y$）および 2 次微分 $(\Delta^2 x, \Delta^2 y)$ を求め，VLA の入力に用いる．これにより，軌道だけでなく動作速度についても反映することを実現する．Sketch Encoder は，スケッチ指示を小規模な MLP で埋め込み，ロボットの状態ベクトルと同様に条件トークンとして Diffusion Transformer へ入力する．

3.3 デモンストレーションデータによるファインチューニング

実機ロボットによるデモンストレーションデータを用いて GR00T N1 をタスクに適応させる．ファインチューニングでは，視覚情報・言語指示・ロボットの状態・スケッチ指示を条件として与え，VLM は固定したまま，条件情報の埋め込み・統合部および Diffusion Transformer を学習する．学習では，教師データである行動系列を生成するように，ノイズ予測誤差を最小化してパラメータを更新する．

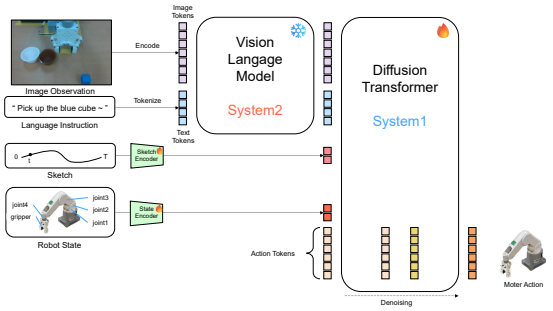


図 2: 提案手法のモデル構造

3.4 データセット作成

提案手法を学習・検証するため，RealSense D435 によって撮影した RGB 動画とロボットの動作ログを同期して記録し，各デモデータに対して言語指示とスケッチ指示を付与することでデータセットを作成する．ロボットは MyPalletizer 260-M5（4 軸＋グリッパ）を使用し，各関節角およびグリッパ開閉量を時刻情報付きで記録する．データセットの作成環境を図 3 に示す．

各デモデータは観測画像列（30fps）と状態・行動（関節角・グリッパ）から構成され，両者を対応付けることで，学習時に同一時刻の各データの参照を可能にする．タスクはピックアンドプレースとし，「青色のキューブを白色のカップに入れる」といった基本的な指示から，「青色のキューブを茶色のカップの前を通って手前側から白色のカップに入れる」などの複雑な指示まで 16 通り設定する．タスクを表す言語指示を各デモデータに付与し，さらに同一な配置に対しても，複数経路のスケッチ指示を作成する．また，1 つのモデルでスケッチ指示の有無の評価を行うため，スケッチ指示を含むデータと含まないデータの 2 種類を学習用に各 300 セット，評価用に 30 セット用意する．

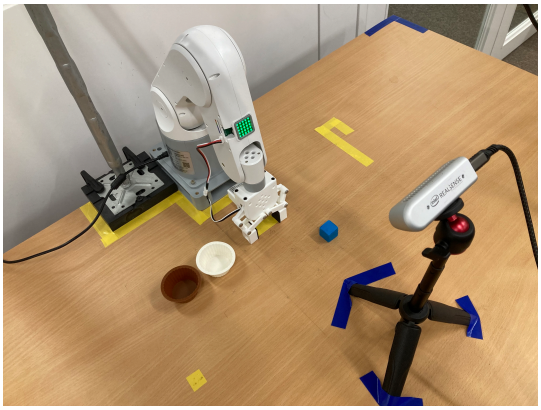


図 3: データセットの作成環境

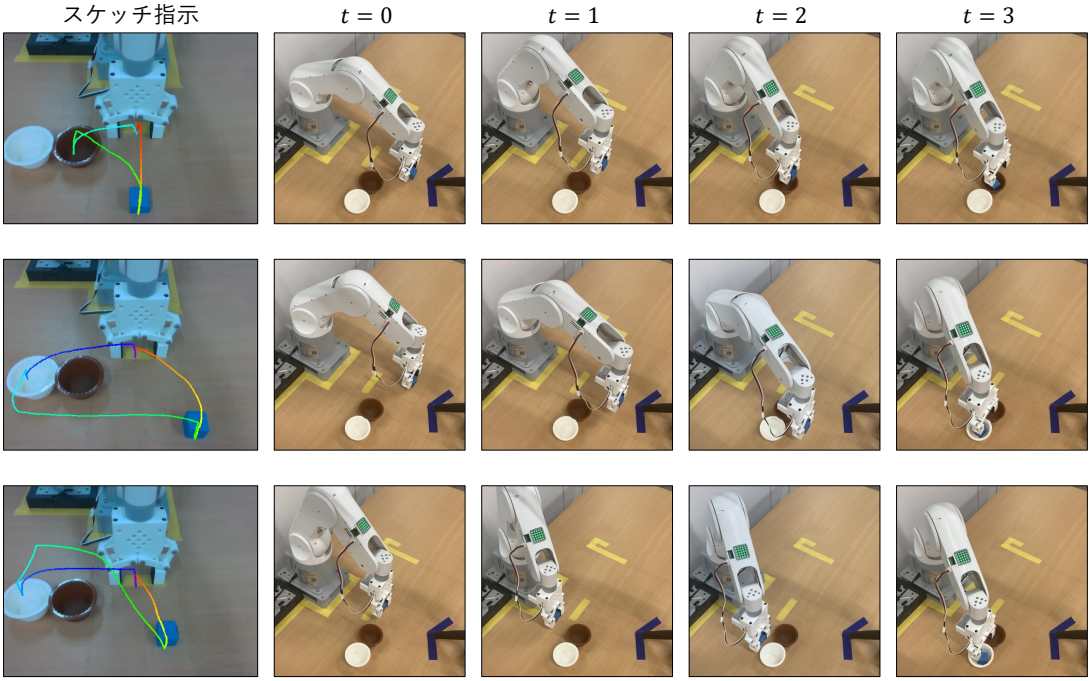


図 4: 定性的評価：スケッチ経路の妥当性

4. 評価実験

作成したデータセットを用いた提案手法の実験を行い，スケッチ指示を用いない場合と用いる場合で比較を行う．学習条件はバッチサイズ 4，学習ステップ 100000，最適化手法 AdamW，学習率 1e-4 とする．定量的評価として実機制御を行った際のタスク成功率を比較し，定性的評価として動作結果を観察することで，経路・速度の妥当性を確認する．

4.1 定量的評価

実機制御におけるタスク成功率を用いて，スケッチ指示の有無による性能差を比較する．ここでタスク成功とは，物体（青色キューブ）を把持し，指示された目標カップに投入できた場合である．カップに投入出来なかった場合や，目標カップに投入できた場合でも，他のカップへの接触や，指示された経路で動作しなかった場合はタスク失敗とする．タスク 1 は直線的に移動するシンプルな経路，タスク 2 は回り込む経路，タスク 3 ではより遠回りの経路や他のカップ位置も考慮した経路とする．各タスクについてスケッチ指示あり／なしの条件でそれぞれ 20 回ずつ検証する．

表 1: 実機実験におけるタスク成功回数

	タスク 1	タスク 2	タスク 3
スケッチあり	20/20	17/20	3/20
スケッチなし	16/20	8/20	0/20

表 1 より，全てのタスクにおいて，スケッチ指示を用いた場合の成功回数が向上した．これにより，スケッチ指示が軌道の意図（直線移動や回り込み方向）を明示し，スケッチ指示なしの場合よりも正確な動作を実現できていると言える．一方で，タスク 3 ではスケッチ指示を用いた場合でも成功回数が 20 回中 3 回のみであった．タスク 3 は，遠回り経路や他のカップ位置の考慮といった複数の制約を同時に満たす必要があり，タスク 1・2 と比較して要求される軌道の多様性が高い．このため，学習データにおけるタスク 3 のバリエーション不足や，スケッチ表現の分解能（点列密度・速度情報）不足により，モデルが安定して意図通りの回避・経路選択を生成できなかったと考えられる．

4.2 定性的評価

経路の差が分かりやすい設定としてタスク 4 を用意し，動作を観察することでスケッチ指示の有無による挙動の差を確認する．まず，スケッチなし条件では，把持から投入までの一連の動作において，目標へ向かう途中で手先が迷うように揺らぐ，直線的に接近して他のカップへ接触する，あるいはカップ手前で停止位置が定まらないといった挙動が観察された．特にタスク 4 のような回り込み動作では，回り込み方向の選択が安定せず，目標カップへ到達できない例が見られた．一方でスケッチあり条件では，移動方向や回り込み方向が明確となり，目標へ向かう経路が安定する傾向が確認できた．

次に，経路の妥当性について評価する．入力したスケッチ指示に対して，手先の移動経路が沿っているかを確認する．図 4 に，可視化したスケッチと，物体把持後からゴールまでの実機の経路の対応例を示す．この結果から，スケッチ指示に沿う経路で動作する様子が確認できた．また，近い経路でスケッチ指示の密度（150 ステップと 300 ステップ）を変えて入力した場合の動作速度の変化も確認できた．

5. おわりに

本研究では，言語指示に基づく VLA モデルにスケッチ指示を時系列の条件情報として入力し，軌道・速度といった具体的な動作意図を行動生成へ反映する手法を提案した．評価では，オフライン指標（MSE）において提案手法の誤差が増加した一方，実機ではスケッチ指示に沿う経路で動作する傾向や，スケッチ指示の点列密度の違いに応じた速度変化が確認できた．今後は，スケッチ指示パターンごとのデータ不足を解消するためのデータ拡充，タスク追加を行い，より高い汎化性能の獲得を実現する．また，別の VLA モデルやロボット実機を使った実験を行い，更なる動作性能の向上を目指す．

参考文献

[1] B. Zitkovich, et al., “RT-2: Vision-Language-Action Models Transfer Web Knowledge to Robotic Control”, CoRL, 2023.

[2] P. Sundaresan, et al., “RT-Sketch: Goal-Conditioned Imitation Learning from Hand-Drawn Sketches”, CoRL, 2024.

[3] J. Bjorck, et al., “GR00T N1: An Open Foundation Model for Generalist Humanoid Robots”, arXiv preprint, arXiv:2503.14734, 2025.

研究業績

野田修平，平川翼，山下隆義，藤吉弘亘，“Transformer モデルを用いたスケッチ指示による把持位置推定”，日本ロボット学会学術講演会，2024.