

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	<b>6.332</b>	0.859	0.708	0.752	0.396	0.290	0.300	<b>0.99</b>
XDA		84.79	6.060	0.857	0.725	0.762	0.440	0.332	0.336	0.97
	✓	<b>87.32</b>	6.123	<b>0.872</b>	<b>0.783</b>	<b>0.802</b>	<b>0.516</b>	<b>0.443</b>	<b>0.426</b>	<b>0.99</b>

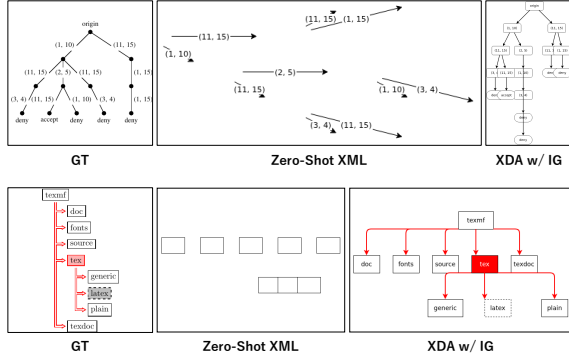


図 2: 各手法により生成された科学図

#### 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言える。

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 Eval-

**Input Query:**  
The diagram is a centralized flowchart with a primary rectangular box in the center labeled 'IPyDrawio' in dark blue. From this central box, there are four arrows extending in four cardinal directions. Above 'IPyDrawio', an arrow points upward to a rectangular box labeled 'Distributing' in orange. From 'Distributing', three arrows extend to three rectangular boxes labeled 'pip', 'conda', and 'npm' from left to right, all in a lighter shade of orange. To the left of 'IPyDrawio', an arrow points leftward to a rectangular box labeled 'User Experience' in red. From 'User Experience', three arrows point to boxes labeled 'familiarity', 'power', and 'robustness', arranged vertically from top to bottom, all in a lighter shade of red. To the right of 'IPyDrawio', an arrow points rightward to a rectangular box labeled 'Q/A' in teal. From 'Q/A', four arrows point to boxes labeled 'acceptance testing', 'unit testing', 'integrity testing', and 'linting', arranged vertically from top to bottom, all in a lighter shade of teal. Below 'IPyDrawio', an arrow points downward to a rectangular box labeled 'Interoperability' in light blue. From 'Interoperability', three arrows point to boxes labeled 'diagrams.net', 'jupyterlab-drawio', 'browsers', and 'jupyterlite', arranged in a T-shape with 'diagrams.net' and 'browsers' to the sides and 'jupyterlab-drawio' and 'jupyterlite' below, all in a lighter shade of blue.

**Visual Evaluator feedback text:**  
1. **"Colors\*"**: The reference image uses a consistent color scheme. The 'IPyDrawio' box should be a dark blue (#1F4E79), the 'Distributing' box should be an orange (#FFB667), the 'User Experience' box should be a red (#FF9999), the 'Q/A' box should be a teal (#87CEFA), and the 'Interoperability' box should be a light blue (#ADD8E6). The subpaths should use a lighter shade of their respective main path colors.  
2. **"Layout and Positioning"**: The layout of the reference image is more organized and balanced. Move the 'Distributing' box directly above the 'IPyDrawio' box and adjust the positions of 'pip', 'conda', and 'npm' so they are evenly spaced horizontally. Similarly, move the 'User Experience' box to the left of 'IPyDrawio' and align 'familiarity', 'power', and 'robustness' vertically below it. The 'Q/A' box should be to the right of 'IPyDrawio', with its subpaths aligned vertically. The 'Interoperability' box should be below 'IPyDrawio', with its subpaths forming a T-shaped arrangement.

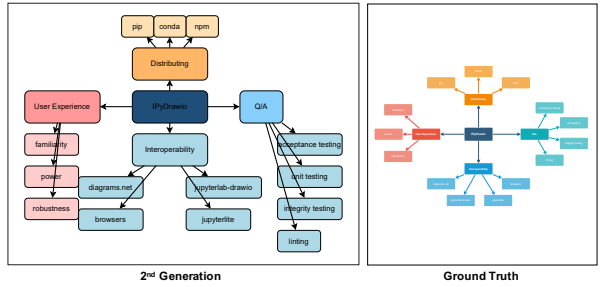


図 3: Visual Evaluator による生成図の変化

uator 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。

#### 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)