

可視化分析における視覚的操作を用いた要約生成システム

松澤 颯太^{1,a)} 伊藤 正彦^{1,b)}

概要: 近年、大規模言語モデル (LLM) を用いた自動要約手法が広く利用されているが、複数の可視化結果比較する探索的分析では、ユーザの分析意図や着眼点が要約に反映しにくいという課題がある。本研究では、複数の可視化画像を比較する分析作業を対象とし、ユーザの視覚的操作と LLM を統合することで、分析意図を反映した要約生成を支援する手法を提案する。提案手法では、Miro 上で行われる画像の配置や囲みといった視覚的操作を分析意図として構造化し、ユーザが行った比較やグループ化を反映した要約文を自動生成する。

1. はじめに

近年、LLM の発展により、自動要約や文章生成は多様な分野で利用されるようになってきている。特に、分析結果の要約や知見整理を自動化する手段として、LLM を用いた支援手法が注目されている [1]。

しかし、複数の可視化結果や画像を比較しながら行う探索的分析においては、要約生成ではモデルが重視する観点が優先されやすく、ユーザの分析意図や着眼点を十分に反映できないという課題がある。また、探索的分析の過程では、ユーザは複数の画像や可視化結果を並べて比較し、配置やグループ化といった視覚的操作を通じて、対象間の共通点や差異を把握する場面が多く見られる。

このような分析では、比較や試行を重ねる中で分析視点が段階的に形成されることが、その判断基準や着目点の多くは明示的なテキストとして表現されない。そのため、入力として与えられるテキスト情報のみに依存する従来の LLM を用いた要約手法では、こうした暗黙的な分析意図や判断基準を十分に取り込むことが難しいと指摘されている [2]。

一方で、複数の画像や可視化結果を同一のキャンバス上に自由に配置できるホワイトボードツールや可視化環境では、ユーザの配置やグループ化といった操作が、比較の視点や着目点を反映していると考えられる。画像の配置や囲みといった操作は、ユーザが類似性や関係性を見出した結果であり、これらは分析結果を理解する上で重要な手がかりとなる。

しかし、視覚的ワークスペースを用いて LLM による要

約生成を支援する研究 [3] は存在する一方で、画像や可視化結果の内容そのものを要約生成の分析対象とすることは行われていない。

そこで本研究では、Miro^{*1} 上で行われる複数の可視化画像を比較する分析作業を対象とし、ユーザが視覚的操作を反復的に行うことで形成される分析意図を構造化し、それを LLM による要約生成に利用することで、分析意図に沿った要約生成を可能にする手法を提案する。

2. 関連研究

LLM を用いた要約生成に関する研究は、近年、単なるテキスト要約にとどまらず、ユーザの分析意図や可視化結果を反映する方向へと拡張されている。Tang ら [3] は、視覚的ワークスペース上でユーザが行う配置やグループ化操作を、LLM による要約生成へ反映する枠組みを提案した。

この研究では、視覚的な整理作業を通じてユーザの解釈や関心を外在化し、それを要約生成に利用することで、分析意図に沿った文章生成を可能にしている。特に、要約生成と視覚的操作を往復する反復的な分析プロセスが重視されており、LLM を「自動要約機」ではなく、意味構築 (sensemaking) を支援する存在として位置づけている点に特徴がある。

一方、ChartInsighter [1] は、時系列グラフを対象とした要約生成において、LLM が生じやすいハルシネーションの問題に着目した研究である。

この研究では、トレンド方向の誤認、極値の誤判断、数値計算誤りなど、時系列チャート要約に特有の誤りを体系的に整理した上で、それらを低減するための手法を提案している。具体的には、外部の数値解析モジュールやマルチ

¹ 北海道情報大学

^{a)} s2221022@s.do-johodai.ac.jp

^{b)} imash@do-johodai.ac.jp

^{*1} <https://miro.com/>

視覚的ワークスペースを用いた要約文生成システム



図 1: 「要約文」表示時のシステム画面

エージェントによる反復的生成、自己一貫性テストを組み合わせることで、要約文の正確性と意味的充実度の両立を図っている。また、ハルシネーションを文単位で注釈したベンチマークを構築し、定量的な評価を行っている点も特徴である。

これらの研究はいずれも、LLM による要約生成を高度化する上で重要な示唆を与えている。しかし、前者は主にテキストやカード型情報を対象としており、時系列グラフのような数値可視化を直接扱うものではない。また、後者は要約文の正確性向上に主眼を置いており、ユーザが可視化を操作しながら分析意図を反映させる過程そのものは対象外としている。

本研究はこれらの先行研究を踏まえ、複数の可視化画像を Miro 上に配置・グループ化することで分析対象を構造化、その結果として形成される視覚的なグループ構造を LLM の入力として直接利用する点に特徴がある。数値データそのものの解析や要約精度の定量評価を目的とするのではなく、ユーザの分析意図と LLM による文章生成を接続する設計例を提示することで、視覚的分析と自然言語生成を統合した分析支援の可能性を示す。

3. 提案手法

提案システムは、Miro 上で構築されたグループ構造をもとに、マルチモーダル入力を用いた LLM による要約生成を支援するものであり、設計は、以下の 4 つの機能要素から構成される。

- グループ構造の抽出
- マルチモーダル情報の統合によるプロンプト生成
- 可視化と要約生成を往復する反復的分析の支援
- 履歴管理とボード再現

本システムは Python を用いて実装し、Miro REST API *2, OpenAI Responses API *3, Dash *4 を組み合わせることで、ユーザ操作を契機として一連の処理が実行される構成とした。

3.1 システム全体像

操作画面には Dash を使い、ブラウザ上で Miro ボードの閲覧と LLM の操作を行える構成とした。画面左側には Miro ボードの表示、右側には GPT へのプロンプト設定、生成結果表示、履歴管理の操作パネルを配置している (図 1)。

画面左側の操作パネルは、以下の 3 つのタブで構成される。

- Miro 現在
最新の Miro ボード状態を表示し、分析対象となる視覚的ワークスペースを直接参照できる。
- Miro 履歴
過去のボード状態を表示し、履歴時点でのグループ構造や配置を確認できる。
- 詳細
画面右側の履歴操作と連動しており、履歴テーブルで特定の実行履歴を選択した際に、対応するテキスト情報を表示するために用いられる。

画面右側の操作パネルは、以下の 3 つのタブで構成される。

- GPT 設定
LLM に渡す入力条件を設定するの画面であり、「プ

*2 <https://developers.miro.com/>

*3 <https://platform.openai.com/docs/api-reference/responses>

*4 <https://dash.plotly.com/>



(a) 要約・提案の方針入力画面



(b) 画像・CSV データ選択画面

図 2: 入力プロンプトを構成するシステム画面



(a) 初期配置

(b) グループ化後の配置

図 3: Miro 上における複数枚の可視化画像の配置例

「プロンプト」「ファイル」のサブタブからなる。要約文および提案生成の方針の構築、参考とする要約文の選択、Miro 上に配置された画像ファイルおよび当該画像の生成に使用された CSV データファイルの選択ができる。

● 生成結果

LLM の出力内容を確認するの画面であり、「要約文」「提案」「入力プロンプト」のサブタブから構成される。生成された要約文および提案の表示と要約生成に用いられた入力プロンプトの要件が確認できる。

● 履歴

過去の実行履歴を管理・再利用するの画面であり、「履歴テーブル」「詳細」のサブタブからなる。生成結果および実行時の Miro ボード状態の履歴が確認できる。

ユーザが「生成開始」ボタンを押下すると、Miro API によるグループ構造の取得と LLM による要約生成が開始される。GPT 設定の入力状態はシステム内に保持され、複数回の操作においても一貫した状態管理が可能である (図 2a)。

3.2 グループ構造の抽出と構造化

まず、複数枚の可視化画像を Miro 上に配置する (図 3a)。2 回目以降は、それまでの要約生成結果を参照しつつ、ユーザが類似した傾向を示すと判断したグループを shape (矩形) で囲むことで、グループ構造を形成する (図 3b)。

なお、ここで示す配置や画像の並べ方は分析手順の一例であり、対象データやユーザの作業方針に応じて柔軟に変化し得る。

囲い込み操作は、ユーザが同一傾向にあると判断した対象を視覚的に明示するものであり、この操作によって得られる構造を LLM の入力として与えることで、ユーザの着眼点を要約生成に反映できる。

提案手法では、Miro REST API を用いて shape, image といったアイテムの種類、座標、サイズを取得する。次に、shape の矩形領域に含まれる image を探索し、各 image に対して、ユーザが Miro 上の shape (矩形) に注釈として入力したグループ名を、tag として付与する。

これにより、Miro 上のグループ構造を JSON 形式で構造化し、LLM が直接参照可能な入力データへ変換する。さらに、グラフ画像は外部 URL から取得し、input-image として LLM に統合する。

3.3 マルチモーダル統合によるプロンプト生成

入力プロンプトは、ユーザの選択に基づいて以下の情報を統合し、LLM に与えるの入力として自動生成される。

- 要約・提案の章立ておよび記述方針を含むシステムプロンプト
- ユーザ入力のプロンプト
- 参考とする要約文 (任意)
- Miro のグループ構造
- 選択された画像 (任意) (図 2b)
- CSV の時系列データ全文 (任意)

このうち、ユーザ入力のプロンプトには、今回の要約生成において重視する指標や時系列変化、複数の shape (グループ) を説明する際の粒度といった要約の方針に加え、提案において点検・改善の対象とする分析条件や構造成方を、自然言語で記述する (図 4b)。

章立てや記述方針はシステムプロンプトによってあらか

【システム指示】
あなたは、複数の可視化結果を比較する分析支援を行う。
必ず次の3つを生成すること。
出力は、Markdown形式とし、見出し名は必ず次の3つを用いること。
要約文
提案
ユーザーのプロンプトに、観点・軸・記述形式・強調点などの指示が含まれる場合は、その指示を優先して反映すること。
ただし、ユーザーの指示があっても上記3つは見出しは省略せず、両方を必ず出力すること。

【要約文の方針】
- 可視化および構造情報から観察される傾向を、客観的・説明的に記述する。

【提案の方針】
- 現在のグループングの妥当性を検証し、追加・新規・再分類の可能性を整理する。
- 要約文を踏まえ、分析の観点整理や比較の着眼点を示す。
- 適度に断定せず、入力情報（構造・データ）に基づく範囲で述べる。

あなたはAIが数値データに置かれているグラフデータアサリトである。
与えられたデータ（構造・軸・記述形式、強調点）を用いて、相手の傾向を多角的に説明するための「summary（要約文）」を生成する。
summary は必ず3つの見出しを用いて、「観察→結果→提案」の順で記述し、必ず以下の3つを用いて記述すること。
要約文
提案
観察

summary の立て方（構造情報）
1. 要約文 (summary)
- ショートで最も重要な傾向を簡潔に3〜4点まとめる。
- 「いつ」「どこで」「何が起きたか」を明確にする。
- Shape (Miro のグループ) のこの要約を要約する。
- 特徴的な傾向 (強調点) を明確に示す。
2. 観察 (observation)
- 各 Shape (グループ) ごとに、傾向、構造、軸、軸ラベル、インジケータ、構造化データの傾向変化を説明する。
- 同一 Shape 内の共通点と差異を整理する。
3. 提案 (recommendation)
- 観察結果を踏まえ、将来的な流れに沿って改善方向を説明する。

suggestion (提案) の立て方
suggestion (提案) では、ユーザーの入力プロンプトが与えられたデータで実行可能な、また観察が明確でない点を指摘する。
必要に応じて、より明確で実行可能なプロンプトへ改善する。
出力は必ず3つを含めなければならない。
① ユーザーの入力プロンプトへの改善
② AI が観察しやすくなるよう構造化した構造化プロンプト

(a) システムプロンプト (抜粋)

(b) ユーザ入力のプロンプト (抜粋)

図 4: LLM に与えた入力プロンプト



(a) データテーブル選択後



(b) 履歴に基づくボード再現後

図 5: 分析履歴の保存および再現を行うシステム画面

はじめ制約されているため (図 4a), ユーザ入力のプロンプトでは、文体や出力形式の指定ではなく、分析対象や注目点に関する補足的な指示を与える役割を担う。

これにより、出力形式の安定性を保ちながら、ユーザの追加指定に応じた柔軟な調整が可能となり、プロンプト構築におけるユーザの負担を軽減する。

生成されたプロンプトは OpenAI Responses API に送信され、マルチモーダル入力として LLM に与えられる。これにより、グループ構造と数値データを同時に参照した要約生成が可能となる。

3.4 反復的分析を支援する生成・再生成フロー

提案手法の特徴は、可視化操作と要約生成を繰り返し行える点にある。ユーザは LLM が生成した要約文および提案を参照しながら、要約文および提案生成の方針や修正や、Miro 上でグループ分けの調整、注釈の追加などを行い、グループ構造の更新ができる。

これらの更新に基づき、システムは再度 Miro のグループ構造を取得し、更新された内容を反映したプロンプトを生成して LLM に渡す。このプロセスを繰り返すことで、要約文および可視化構造の完成度を徐々に高めることが可能となる。

提案は、ユーザによる提案の方針に基づき、可視化構造および要約の方針についての検討を補助する形で生成される。これにより、ユーザの試行錯誤による改善を支援するとともに、分析結果の完成度向上に寄与する。

これら 2 つの要素を組み合わせることで、ユーザによる

主体的な分析視点と、LLM による補助的な視点とを統合した分析が可能となり、可視化構造および要約結果の完成度をより高い水準へと導くことができる。

3.5 履歴管理とボード再現による再現性確保

生成時に用いられた Miro のグループ構造、画像、CSV、プロンプト、生成結果、モデル種別、トークン使用量は履歴として保存される。データテーブルから履歴を選択すると、対応する要約文や関連するテキスト情報が即座に表示され、過去の生成内容を容易に参照できる (図 5a)。

さらに、保存された snapshot を用いて、生成時点の Miro ボードを再現する機能を備える。再現処理では、対象ボード上のアイテムを一度削除した後、shape, text, image を位置やスタイルとともに再構成することで、当時の視覚的配置を再現する。この機構により、分析過程の可視性と再現性が確保される (図 5b)。

これにより、過去の生成結果を単なる記録として残すのではなく、後続の分析や再生成のための参照情報として活用できる。

4. 探索的分析事例と考察

4.1 探索的分析事例

本章では、提案手法を用いた探索的分析の一例を示す。本事例は、手法の有効性や生成内容の正確性を評価することを目的とするものではなく、システムの利用形態と生成結果の特徴を具体的に示すためのものである。

使用する可視化画像は、MLB が公開する Baseball Sa-

