

卒業論文

一点物売買プラットフォームの動作検証を目的とした社会的シナリオに則ったデータ生成エンジエン トの開発

2025 年度

指導教員 植松 幸生 准教授

(6322087)

三笠 悠太郎

東京理科大学 創域理工学部 情報計算科学科

提出日： 2026 年 2 月 15 日

一点物売買プラットフォームの動作検証を目的とした社会的シナリオに 則ったデータ生成エージェントの開発

要旨

近年，クリエイターエコノミーの発展に伴い，作品単体の価値だけでなく，その背景にあるストーリーやクリエイターとファンが形成するコミュニティの文脈を統合して流通させる Web プラットフォームの重要性が高まっている。このようなプラットフォームにおいて，ユーザー間の社会的相互作用や共同制作プロジェクトといった関係性を記録・表現する必要があるため，そのデータベースは社会的シナリオを反映した複雑なリレーション構造となる傾向にある。このようなプラットフォームの開発初期段階において，UI 表示や検索システムの有用性を検証するためには，サービス固有のデータ構造と整合性を保った大量のテストデータが不可欠である。しかしながら，社会的シナリオを反映したデータを手動で大量に用意することは困難であり，従来のランダムなダミーデータ生成手法では，リレーションの整合性やコンテンツの意味的な文脈を再現できないという課題があった。

本研究では，大規模言語モデル (LLM) を用いて，ユーザの社会的背景や文脈を記述した社会的シナリオを構築し，それに基づいてサービス固有のリレーションを持つデータ群を生成するエージェントを開発した。本エージェントは，LangGraph を用いたステートマシンとして設計されており，データベースを活用した記憶機構を実装している。これにより，過去に生成したデータと重複しない多様なコンセプトを立案し，そのコンセプトに基づいて相互に整合性の取れたユーザー，プロジェクト，作品群をひとまとめの単位として，本プラットフォーム上のデータ構造に特化した，自律的にデータを生成・投入する能力を持つデータ生成エージェントを開発した。

提案手法の有効性を検証するため，実験環境として一点物取引に特化した Web プラットフォームを開発し，google, Open AI の 4 つの基盤モデルに対して本エージェントを適用して 1000 件のデータ生成実験を行った。評価の結果，過去の生成履歴を参照する記憶機構を導入することで，Gemini 3 Flash にて生成される語彙数が 39.0% 増加し，GPT-5.2においては全生成データのコサイン類似度分布が効果量 $d = -0.51$ で低下した。これらの結果は，記憶機構が生成内容の重複や特定のパターンへの収束を抑制し，生成されるデータセット全体の多様性を向上させることを示している。本研究は，LLM を用いた社

会的シナリオ生成が、開発初期段階においても実運用環境に近い整合性と多様性を持つ
テストデータの生成に有効であることを実証した。

Title
— **Subtitle**

Summary

The author shall describe his/her thesis in English text. 150 words.

Contents

要旨	i
Summary	iii
1 序論	1
1.1 背景	1
1.2 開発上の課題：複雑な文脈を持つシステムの検証難易度	3
1.3 本研究の目的	3
2 関連研究	4
2.1 ソフトウェアテストデータの生成手法	4
2.2 統計的・機械学習的なデータ合成	5
2.3 LLM を用いたリレーションナルデータの構築・操作	5
2.4 社会シミュレーションと生成的エージェント	6
3 提案手法：社会的シナリオに基づくデータ生成エージェント	6
3.1 アプローチの概要	6
3.2 システムアーキテクチャ	7
3.3 社会的シナリオのデータモデル	8
3.4 記憶機構と RAG	10
3.5 プロンプトエンジニアリング	10
4 実験環境：一点物売買プラットフォーム	11
4.1 プラットフォームの概要	11
4.2 システム構成	11
4.3 データモデル設計	15
4.4 Project 管理方式	17
4.5 統合型検索システム	19
4.6 販売形態と価格決定アルゴリズム	20
4.7 テストデータに対する要件と課題	23
5 評価実験	24
5.1 実験設定	24
5.2 生成データの実例	25
5.3 語彙数の増加推移	33
5.4 ベクトル類似度による重複検知	34
5.5 LLM Judge による定性評価	36
5.6 数値的類似度と意味的多様性の関係	37
5.7 まとめ	39

6

結論

39

謝辞

41

1 序論

1.1 背景

1.1.1 クリエイターエコノミーの拡大と作品流通の変容

近年，インターネット技術の発展とソーシャルネットワーキングサービスの普及により，個人が自身の創作物を発信・販売し，収益を得るクリエイターエコノミーが急速に拡大している [1]。動画投稿，ライブ配信，ハンドメイド作品の販売，デジタルコンテンツの提供など，その形態は多岐にわたり，従来は企業や専門家が独占していた生産・流通・販売のプロセスが個人レベルまで民主化された。

特に，物理的な創作物を制作するクリエイターにとって，Mercari [2] や minne [3] といった CtoC マーケットプレイスの普及，あるいは BASE [4] のように個人が手軽に独自のオンラインストアを開設できるサービスの登場は，販路開拓のハードルを劇的に引き下げた。中でも，イラストや同人誌などのサブカルチャー領域に特化した BOOTH [5] は，物理的な作品とデジタルコンテンツを並列して扱うことができ，本研究が対象とするような「作品の文脈」を重視するクリエイター層から厚い支持を得ている。これにより，趣味の延長線上で制作を行っていた個人が，容易に売り手として市場に参加できる環境が整った。

しかし，これらの既存プラットフォームの多くは，既製品や大量生産品，あるいは不用品の流通に最適化されて設計されている側面が強い。商品自体は代替可能な商品として扱われ，その背後にある制作者の思想，制作過程の試行錯誤，作品が生まれた文脈は，商品説明欄の補助的なテキスト情報として扱われるに留まる。

一点物や極めて少数の生産数しか持たない創作物において，その価値の源泉は物理的な素材や機能性よりも，この物語にこそ存在する。既存のプラットフォームでは，こうした作品固有の物語性や文脈を十分に表現・保存することが難しく，結果として作品が単なる商品として消費され，制作者のプランディングや持続的なファンコミュニティの形成に繋がりにくいという課題がある。

1.1.2 物理的な展示・即売会における制約とアーカイブの欠如

デジタルプラットフォームの対極として，学園祭，展示会，即売会といった物理的な場での作品発表も依然として重要な役割を果たしている。これらの場は，制作者と鑑賞者が直接交流し，作品の熱量を共有できる貴重な機会である。

しかし、物理的なイベントには一過性という避けられない制約が存在する。第一に、展示情報のアーカイブ化の問題である。展示会のために制作された解説パネルやキャプションボードは、イベント終了とともに廃棄または散逸するが多く、そこに記された詳細な作品情報はデジタル空間に蓄積されない。結果として、イベントに参加できなかつた潜在的なファンには作品の魅力が届かず、制作者のポートフォリオとしても十分に機能しない場合が多い。

第二に、金銭授受と販売機会の損失である。学園祭や小規模な展示会では、販売機能を持たず展示のみとするケースや、販売する場合でも現金取引のみの対応となるケースが多い。また、展示されている作品そのものを購入したいという需要があつても、会期終了後の引き渡し手続きや配送手配の煩雑さから、その場での成約に至らないことも頻繁に発生する。このように、物理的な展示空間とデジタル上の決済・物流システムが分断されることにより、多くの経済的機会が損失しているのが現状である。

1.1.3 一点物取引における価格決定と信頼性の課題

一点物、あるいは極めて供給量の少ない作品の流通においては、価格決定メカニズムと取引の信頼性に関して特有の課題が存在する。

まず、価格決定においては、需要が供給を大幅に上回る場合、定価販売形式では早い者勝ちとなり、転売目的の購入や、botによる自動購入などの問題を引き起こしやすい。これに対し、オークション形式は需要に応じた適正価格を発見する有効な手段であるが、既存のオークションサイトは、前述の通り不用品処分の文脈が強く、クリエイターが自身の作品を発表・販売する場としてのプランディング機能に欠ける場合がある。

1.1.4 共同体による創作活動と収益分配の複雑性

創作活動は必ずしも個人だけで完結するものではない。サークル活動、合同誌、アンソロジー企画、YouTuber 同士のコラボレーションなど、複数のクリエイターが関与するプロジェクト単位での活動も一般的である。しかし、既存の CtoC プラットフォームの多くは一人の売り手対一人の買い手という単純な取引モデルを前提としている。そのため、一つの作品またはイベントから得られた収益を、関与した複数のメンバーに分配する場合、代表者が一度全額を受け取り、その後銀行振込などで個別に再分配するという、極めてアナログで煩雑な事務作業が必要となる。これは、企画主催者の負担を増大させ、継続的な共同制作活動を阻害する要因となっている。

1.2 開発上の課題：複雑な文脈を持つシステムの検証難易度

前述したような作品の物語や複雑な権利・収益構造を持つプラットフォームを新規に開発する際、最大の障壁となるのが検証用データの不足である。

一般的なECサイト（商品と価格のみの管理）であれば、ランダムな文字列やダミー画像を用いた機械的なデータ生成（Faker.js [6] 等）で十分に機能検証が可能である。しかし、本研究が対象とするような文脈を重視するシステムにおいては、以下の理由により、既存のテストデータ生成手法が通用しない。

1. リレーションの整合性：プロジェクト主催者と参加メンバー、そして出品された作品の間には、矛盾のない論理的な関係性が求められる。ランダム生成では、「部外者が他人のプロジェクトの作品を管理している」といった矛盾が頻発し、権限管理ロジックのテストが成立しない。
2. 検索・推薦アルゴリズムの検証：文脈に基づく検索システムを評価するには、データ自体に意味のあるテキスト（制作背景や動機）が含まれている必要がある。Lorem Ipsumのような無意味なテキストでは、検索精度の検証が不可能である。

したがって、開発者は「手動で時間をかけて整合性のあるデータを作る」か、「質の低いデータで妥協して検証を行う」かの二者択一を迫られており、これが開発効率と品質向上のボトルネックとなっていた。

1.3 本研究の目的

本研究の目的は、一点物売買プラットフォームの開発と、それを支える新たなテストデータ生成手法の確立の2点にある。

1.3.1 一点物作品のナラティブを保存するプラットフォームの開発

第一に、既存のEコマースでは埋没しがちな作品の文脈を構造化データとして保存・活用できる、新たなCtoCプラットフォームを構築する（第4章にて詳述）。これは本研究において、提案するデータ生成手法の有効性を検証するための実験環境としての役割も担う。

1.3.2 LLM を用いた社会的シナリオ生成によるシステム検証

第二の，そして本論文の主眼となる目的は，上記のような複雑なリレーションを持つWeb サービスの開発において，大規模言語モデル（LLM）を活用した社会的シナリオに基づくテストデータ生成手法を提案・実証することである。

本研究では，LLM に「架空のユーザーの人格」や「プロジェクトの動機」といった社会的背景（社会的シナリオ）を与え，自律的にデータベースへの登録を行わせるエージェントシステムを構築する。さらに，エージェントに記憶機構（RAG）を組み込むことで，既存データとの衝突を回避し，多様なシナリオを生成できるかを実験的に検証する。これにより，学習データが存在しない環境下においても，あたかも人間が実際に活動しているかのような，高密度で整合性の取れたテストデータを大量生成できることを示す。

2 関連研究

本章では，ソフトウェアテストデータの生成手法，大規模言語モデル（LLM）のデータ生成への応用に関する先行研究を概観し，本研究の位置づけを明確にする。

2.1 ソフトウェアテストデータの生成手法

Web アプリケーションの開発現場において，動作検証や負荷試験のためのテストデータ生成は不可欠な工程であり，主に構文的整合性やコード網羅率の確保に主眼が置かれてきた [7]。

代表的なアプローチとして，Faker.js [6] や JavaFaker [8] に代表されるライブラリを用いたランダム生成手法が存在する。これらは，正規表現や定義済みの辞書に基づき，氏名，住所，メールアドレスといった形式が決まったデータを大量に生成することに長けている。また，Mockaroo [9] のようなデータ生成サービスでは，GUI 上でスキーマ定義を行うことで，ある程度の外部キー制約を満たした CSV や SQL データを生成可能である。さらに，近年の Web 開発フレームワーク（例：Prisma [10] などの ORM）には，開発初期のシードデータ（初期データ）を投入する機能が標準で備わっていることが多いが，これらも基本的には開発者が手動で作成した固定データか，あるいは前述の Faker 等で生成したランダムデータを投入するための機構に留まる。

しかし，これらの手法は，本来アプリケーションのログイン機能や権限管理といった基本機能の動作検証を目的としたものであり，データ間の複雑な意味的整合性や文脈を再現

することを主眼としていない。例えば、「居住地が日本であるにも関わらず住所がニューヨークに設定されている」といった整合性の欠如や、「ユーザー間のメッセージ交換に見られる時系列的な因果関係」を含む複雑なシナリオを再現することは、単純なランダムサンプリングでは原理的に困難である。本研究が対象とする CtoC プラットフォームのような、ユーザー間の相互作用や作品の背景ストーリーが重要となるシステムの検証において、これらの既存手法は不十分であると言える。

2.2 統計的・機械学習的なデータ合成

統計モデルや機械学習を用いて、既存データセットの分布を学習し、類似した特性を持つデータを新たに生成する手法も広く研究されている。代表的な手法として、条件付きGAN を用いた CTGAN [11] や、リレーションナルデータベース全体の生成に特化したベンチマークである SyntheRela [12] において評価されている SDV (Synthetic Data Vault) などが挙げられる。

これらの手法は、プライバシー保護（匿名化）やデータ拡張（アップサンプリング）において強力なツールとなるが、適用するには「学習元となる大量の既存データ」が存在することが前提となる。したがって、新規サービスの立ち上げ時のような「学習データが全く存在しない状態」においては適用することができない。本研究は、学習データが存在しないフェーズにおいて、LLM の知識と創造性を利用してゼロからデータを生成するアプローチをとるため、これらの統計的手法とは適用領域が根本的に異なる。

2.3 LLM を用いたリレーションナルデータの構築・操作

近年、LLM の高度な言語理解能力を活かし、複雑なスキーマ構造を持つリレーションナルデータを扱う研究が進展している。

Li ら [13] の OmniSQL は、Text-to-SQL モデルの学習用データ不足を解消するために、Web 上の表データから多様なデータベーススキーマと SQL クエリを自動合成するフレームワークを提案した。また、Sadia ら [14] の SQuID は、非構造化テキストからリレーションナルデータベースを構築するタスク (Text2R) において、LLM に直接 SQL を書かせるのではなく、スキーマ設計とデータ投入のプロセスを分離することで整合性を高める手法を示した。さらに、Kim ら [15] の ReFuGe は、予測タスクの精度向上のために、LLM エージェントを用いてリレーションナルデータベースから有効な特徴量を自動生成す

る枠組みを提案している。

これらの研究は、LLM が複雑なリレーション構造を理解・操作できることを示しているが、その主目的はあくまで「学習データの作成」や「既存情報の抽出・分析」にある。本研究は、これらの研究と技術基盤（LLM エージェント、プロセス分解アプローチ）を共有しつつ、目的を「架空の社会的シナリオに基づく新規データの創造」に置く点で独自性がある。

2.4 社会シミュレーションと生成的エージェント

LLM を用いて人間らしい振る舞いをシミュレートする研究として、Park ら [16] の Generative Agents が挙げられる。彼らは、記憶と計画を持つ複数のエージェントを仮想空間内で生活させ、エージェント同士の創発的な社会的相互作用（例：パーティーの企画）が生まれることを実証した。

この「記憶に基づく一貫した行動生成」という概念は本研究の基礎となっているが、Generative Agents の出力は自然言語のログ（非構造化データ）であり、そのままではリレーションナルデータベースの厳密な制約（外部キー・データ型）を満たすテストデータとして利用できない。本研究の貢献は、Generative Agents のような「人間らしい文脈や動機」を維持しつつ、それを SQuID のような「構造的整合性」を持つリレーションナルデータとして出力するためのシステムアーキテクチャを提案する点にある。

3 提案手法：社会的シナリオに基づくデータ生成エージェント

本章では、一点物売買プラットフォームのような複雑なリレーション構造を持つシステムに対し、整合性と多様性を両立したテストデータを自律的に生成するエージェントシステムについて述べる。

3.1 アプローチの概要

従来、ウェブサービスの開発におけるテストデータ生成には、Faker.js 等のライブラリを用いたランダム生成や、固定のダミーデータセットの流用が一般的であった。しかし、これらの手法では「ユーザー A がプロジェクト B を立ち上げ、その文脈に沿った作品 C を制作し、ユーザー D がそれを購入する」といった、エンティティ間の意味的な整合性（社会的シナリオ）を再現することは困難である。本研究では、大規模言語モデル（LLM）

を用い、データ生成を複数段階に分解した生成プロセスを設計することで、上記の課題に対処する。提案するエージェントは、単一テーブルの値を個別に生成するのではなく、ユーザー、プロジェクト、作品(Item)の関係を含む「社会的シナリオ」を最小単位として生成する。これにより、生成データが外部キー制約等の技術的整合性を満たすだけでなく、文脈的一貫性を持つようとする。

3.2 システムアーキテクチャ

本エージェントは、LangGraph [17] を用いたステートマシンとして実装されている。LangGraph は、LLM を含む処理の制御フローをグラフ構造(ノードとエッジ)として定義するフレームワークであり、Yao ら [18] が提案した ReAct パターン等の推論プロセスを表現できる。本研究では、処理を役割ごとに分割したノード(Director Agent, Scenarist Agent, Designer Agent, Saver Agent)として定義し、共有状態を更新しながらパイプライン処理を行うアーキテクチャを採用した。図 1 にシステム全体の構成を示す。

3.2.1 ステートマシンの構成

エージェントの処理フローは以下の 4 つの主要ノードによって構成される。

1. Director Agent: 生成タスクの発行と分布(カテゴリバランス)の管理を担当する。過去の生成ログを参照し、現在不足しているカテゴリや構造タイプを特定する。本ノードは LLM を使用せず、あらかじめ定義された確率モデル(Distribution Matrix)に基づき、次に生成すべきカテゴリと構造タイプの組を決定し、下流の Designer Agent へ指示を発行する。
2. Designer Agent: Director Agent から渡された抽象的な枠組みに基づき、具体的な「シナリオのコンセプト」を立案する。この段階で記憶機構(RAG)を活用し、過去に生成された類似シナリオとの重複を避けるよう、テーマや記述方針を決定する。コードリスト 1 に、本エージェントのシステムプロンプトを示す。

```
1 あなたはテストデータ生成のための「構成作家」です。
2 指定された「カテゴリ」と「構造タイプ」に基づいて、具体的でユニークなシナリオのコンセプトを立案してください。
3
4 ## プラットフォームの定義
5 このプラットフォームは、現実世界のユーザーが、自分の制作した作品(模型、手芸、イラストなど)やコレクションを展示・売買するためのサービス( )です。CtoCArtSquare
6 作中の登場人物や兵器そのものを生成するのではなく、「それを趣味として楽しむ人々」のシナリオを作成してください。
7
```

```

8    ## 指示
9    - 既存のシナリオとは異なる視点、設定、トーンで企画してください。
10   - ニッチな設定や、エッジの効いた設定を歓迎します。
11

```

Listing 1 Designer Agent (コンセプト立案) のシステムプロンプト

- Scenarist Agent: Designer Agent が作成したコンセプト指示書に基づき、具体的なデータ (User, Project, Item) を生成する。Structured Output 機能を用いて、テキスト記述の生成とデータベーススキーマ (JSON Schema) へのマッピングを同時に行う。これにより、文脈的整合性と技術的整合性 (外部キー制約等) の両立を図る。コードリスト 2 に、本エージェントのシステムプロンプトを示す。

```

1 あなたは架空のクリエイターブラットフォーム(ArtSquare)のデータを生成する「脚本家」です。
2 与えられた「カテゴリ」と「社会構造(Structure Type)」に基づいて、
3 整合性の取れた一つの小さな社会(シナリオクラスター)を生成してください。
4
5 ## 重要な前提
6 - はの創作・ホビー・コレクションのプラットフォームです。ArtSquareCtoC
7 - 生成されるは、現実的に個人が売買・展示できるもの(模型、手芸、同人誌、素材、データなど)に限られます。
Item
8
9 ## 制約事項
10 - User, Project, Item の各データを作成すること。
11 - tempId を使用して、生成データ内で矛盾のないリレーションを構築すること。
12 - は形式で、読み応えのある記事のように書くこと。DescriptionMarkdown
13 - や説明文の口調は、設定された「ペルソナ」や「カテゴリの文化」に合わせること。Bio
14

```

Listing 2 Scenarist Agent (データ実装) のシステムプロンプト

- Saver Agent: 生成された構造化データをデータベースに保存すると同時に、その内容をベクトル化 (Embedding) し、エージェントの長期記憶に格納する。これにより、次の Designer Agent の推論時に参照可能な状態とする。

3.3 社会的シナリオのデータモデル

本エージェントが生成するデータの最小単位である「シナリオクラスター」は、以下の 3 層構造を持つ。この構造は、実験対象とするプラットフォーム (第 4 章で詳述) のデータモデルと 1 対 1 で対応している。

3.3.1 Concept

すべてのデータの起点となる「テーマ」である。ここには、活動のジャンル (例: Handcraft, Gundam), 動機 (例: リサイクル, 教育, 偏愛), および雰囲気 (例: Cyberpunk,

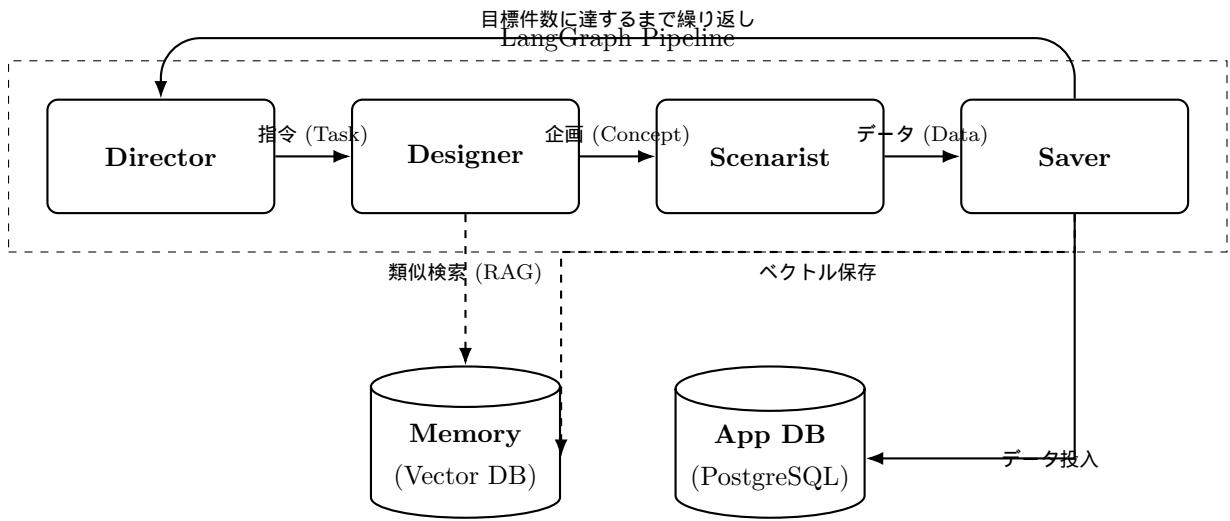


Figure1 社会的シナリオ生成エージェントのシステムアーキテクチャ

Nostalgic) が含まれる . RAG による重複排除は , 主にこの Concept レベルで行われる .

3.3.2 Community

Concept を実現するために集まった人々の定義である .

- Organizer: プロジェクトの主催者 . 強い動機とリーダーシップを持つ .
- Collaborator: 協力者 . 技術提供や資材提供など , 主催者を補完する役割を持つ .

LLM は , これらの登場人物に対し , 性格特性や背景ストーリーを付与し , User プロフィールの自己紹介文などを生成する .

3.3.3 Output

コミュニティ活動の結果として生み出された「作品」である . 各作品には , 以下の要素が付与される .

- Narrative: その作品がなぜ作られたかという背景 .
- Spec: 素材 , サイズ , 重量などの物理的特性 .
- Log: 制作過程の記録や , 使用されたツールの履歴 .

3.4 記憶機構と RAG

本研究では、エージェントが過去の生成履歴を記憶し、それを参照しながら次の生成を行う機構を導入する。これにより、生成数が増加した場合でも、類似したコンセプトの反復を抑制することを狙う。

3.4.1 ベクトル検索による重複検知

記憶機構の実装には、軽量かつ高速なベクトル検索ライブラリである `sqlite-vec` を採用した。エージェントが新たなシナリオを生成する際、以下のプロセスを実行する。

1. Director Agent: 次に生成すべきカテゴリと構造を決定し、タスクを発行する。
2. Designer Agent: タスクに基づき、ベクトルデータベースから類似する過去のシナリオ（上位 N 件）を検索する。
3. 検索された既存シナリオを Prompt に含め、「これらとは異なる、新規の観点を持つシナリオを作成せよ」という制約のもと、新しいコンセプトを立案する。
4. Scenarist Agent: 確定したコンセプトに基づき、詳細なデータを実装する。

この RAG ループにより、エージェントは過去の生成物と類似した出力の反復を避けるように振る舞うことが期待される。第 5 章の評価では、語彙数や類似度指標の観点から、本機構の導入による差を検証する。

3.5 プロンプトエンジニアリング

LLM に対して、実験対象となるプラットフォームの想定に沿ったデータを生成させるために、以下のプロンプト戦略を採用した。

3.5.1 Few-Shot Prompting with Anti-Patterns

良質な例だけでなく、避けるべき例を明示的に与えた。特に、「LLM が生成する傾向にある典型的なパターン」を禁止事項として列挙した。

- 禁止用語：「革新的な」「ソリューション」「多様なニーズ」「シームレスな」等的一般的なマーケティング用語。
- 推奨スタイル：個人の嗜好や生活様式に焦点を当てた記述。

3.5.2 Persona Adoption

各ノードにおいて、LLM に明確なペルソナを与えることで、出力のトーンを制御した。例えば Designer Agent には、データ生成の方針や制約を意識した役割を与え、出力の文体や内容の偏りを抑制することを狙った。

4 実験環境：一点物売買プラットフォーム

本研究では、提案するデータ生成エージェントの有効性を検証するための実験環境として、一点物の取引に特化した Web プラットフォームを開発した。本章では、このプラットフォームの概要、技術構成、およびテストデータに求められる要件について述べる。

4.1 プラットフォームの概要

開発したプラットフォームは、クリエイターが制作した「作品」を制作背景とともに販売・購入できる CtoC マーケットプレイスである。本プラットフォームでは、作品の説明文に加えて制作背景や制作過程の記録（制作ログ）を保存・提示できることを重視し、作品単体の情報だけでなく、活動の文脈も含めて閲覧できるように設計した。

主な特徴は以下の通りである。

- プロジェクトベースの活動：ユーザーは単独で出品するだけでなく、プロジェクトを立ち上げ、複数人で協力して制作・販売を行うことができる。
- 制作ログの重視：作品の完成形だけでなく、使用した素材、制作中の失敗、試行錯誤の履歴をログとして記録し、作品価値の一部として提示する。
- 複雑な権利・収益分配：一つの作品に対し、素材提供者、制作者、プロジェクト主催者など複数のステークホルダーが存在し、売上は Stripe Connect を通じて自動的に分配される。

4.2 システム構成

本プラットフォームは、フロントエンド/バックエンド/データベース/決済基盤を分離した構成で実装した（図 2）。

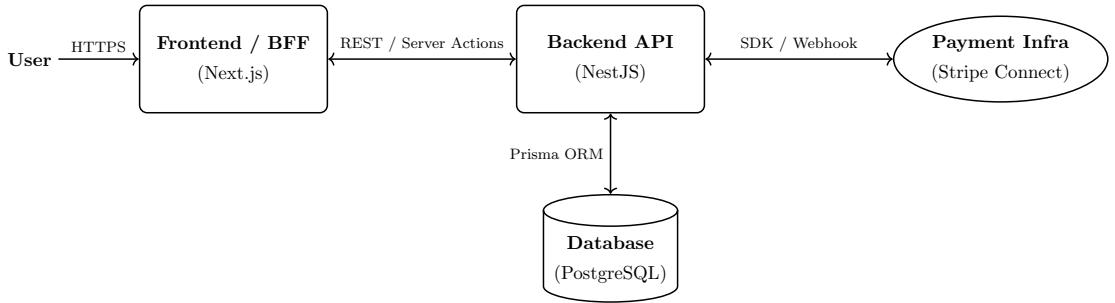


Figure2 Payment Platform 2 のシステム構成

4.2.1 フロントエンド

ユーザーとの接点となるフロントエンドには、React ベースのフレームワークである Next.js (App Router) [19] を採用した。本システムでは、単なる画面描画だけでなく、サーバーサイドでのデータフェッチや認証セッション管理を行う BFF としての役割も担わせている。

ディレクトリベース・ルーティングの採用 ソースコード構成 (src/app 配下)において、ファイルシステムの階層構造をそのまま URL パスにマッピングするディレクトリベース・ルーティングを採用した。例えば、クリエイターのポートフォリオページは src/app/[userPubId]/page.tsx、作品詳細ページは src/app/[userPubId]/[itemPubId]/page.tsx として配置している。このように動的パラメータをディレクトリ名に含めることで、ユーザーおよび作品の公開識別子を URL に反映したページ構成を実装できる。

Server Actions とデータ取得の最適化 コンポーネント設計においては、React Server Components (RSC) を全面的に採用した。作品情報 (Markdown テキスト) やユーザープロファイルなどの静的な初期データは、サーバーサイドでデータベース (またはバックエンド API) から直接取得し、HTML としてクライアントに送信される。また、フォーム送信や入札アクションなどの動的な操作には Server Actions (src/actions/) を用いることで、クライアントサイドに露出する API エンドポイントを隠蔽しつつ、型安全な関数呼び出しとしてバックエンドとの通信を実装している。

認証と認可 認証基盤には Node.js のライブラリである NextAuth.js (src/api/auth/[...nextauth]) を導入し、Google OAuth およびメールアドレス認証によるログイン

フローを実装した。セッション情報は JWT (JSON Web Token) として発行し , HTTP Only Cookie を用いて管理する。

4.2.2 バックエンド

システムのコアロジックを担当するバックエンドには , Node.js フレームワークである NestJS [20] を採用した。TypeScript による静的型付けと , 依存性の注入 (DI) によるモジュール構成を利用し , 機能ごとに責務を分割した。

モジュール分割による関心の分離 アプリケーションは機能単位でモジュールに分割されている。

User Module アカウント管理 , プロフィール更新

Auth Module 認証処理

Item Module 作品の CRUD , Markdown 解析 , 在庫管理

Order Module 注文作成 , 決済ステータスの遷移管理

Project Module 企画のガバナンス管理 , 収益分配計算

このようにドメインごとに責務を分離し , 機能追加や変更の際に影響範囲を局所化しやすい構成とした。

非同期イベント処理 本システムでは決済処理を Stripe に委譲しているため , 決済の成功・失敗や , オークションにおける与信枠の確保の結果は , すべて非同期の Webhook イベントとして通知される。これを受け取るための専用のエンドポイント (src/webhook) を実装し , Stripe からの署名検証を行った上で , Order データベースのステータスを更新する。この処理により , 決済の結果を非同期イベントとして取り込み , 状態遷移を管理する。

4.2.3 データベース

データの永続化には , リレーションナルデータベースである PostgreSQL [21] を採用した。複雑な多対多のリレーション (例 : Project と Collaborators , Item と Bids) を矛盾なく管理し , ACID 特性 (原子性 , 一貫性 , 独立性 , 永続性) を備えたトランザクション処理を実現するためである。

Prisma ORM による型安全なデータアクセス アプリケーション層とデータベース層の架け橋として Prisma [10] を使用している .schema.prisma ファイルに定義されたデータモデルに基づき、型定義ファイル (TypeScript 型) が自動生成される。これにより、バックエンドの開発において「存在しないカラムへのアクセス」や「型の不一致」といった単純な誤りをコンパイル時に検出しやすくなる。

4.2.4 決済および送金基盤 (Stripe Connect)

CtoC プラットフォームにおける金銭取引の決済インフラには、Stripe Connect [22] を採用した。本システムでは、売り手 (クリエイター) への送金管理、本人確認、およびオーバークション等の決済フローに対応するために、Stripe が提供する機能を利用している。

連結アカウントによるユーザー管理 プラットフォームに参加するクリエイター (売り手) は、Stripe の「連結アカウント」として管理される。User モデルの stripeAccountId カラムには、Stripe 側で発行された一意のアカウント ID (acct...) が保存される。これにより、売上の入金先口座の設定や本人確認の手続きを Stripe 側の機能として取り扱い、プラットフォーム側では識別子の管理と状態の参照を行う。

支払いと送金の分離 資金フローにおいては、買い手からの支払いを即時に売り手へ送金するのではなく、支払いと送金を分離する方式を採用した。この方式では、買い手の決済は一度プラットフォームのアカウントに対して行われ、その後、任意のタイミングで売り手の連結アカウントへ送金が実行される。これにより、以下の要件を満たしている。

エスクロー機能 商品の発送完了や受け取り評価が行われるまで、プラットフォーム上で売上を一時的に保持し、トラブル時の返金を容易にする。

収益分配の柔軟性 プロジェクト主催者への手数料やプラットフォーム手数料を差し引いた上で、正確な金額をクリエイターに分配する。

オークションにおける与信枠の活用 イングリッシュオークションや抽選販売において、当選後の未払いは課題の一つである。本システムではこれを抑制するため、クレジットカードのオーソリゼーション (与信枠確保) 機能を活用した決済フローを構築した。

与信枠確保 入札や抽選応募の時点で PaymentIntent を作成し、capture_method: 'manual' オプションを用いてカードの与信枠のみを確保する。この段階では実際の請求

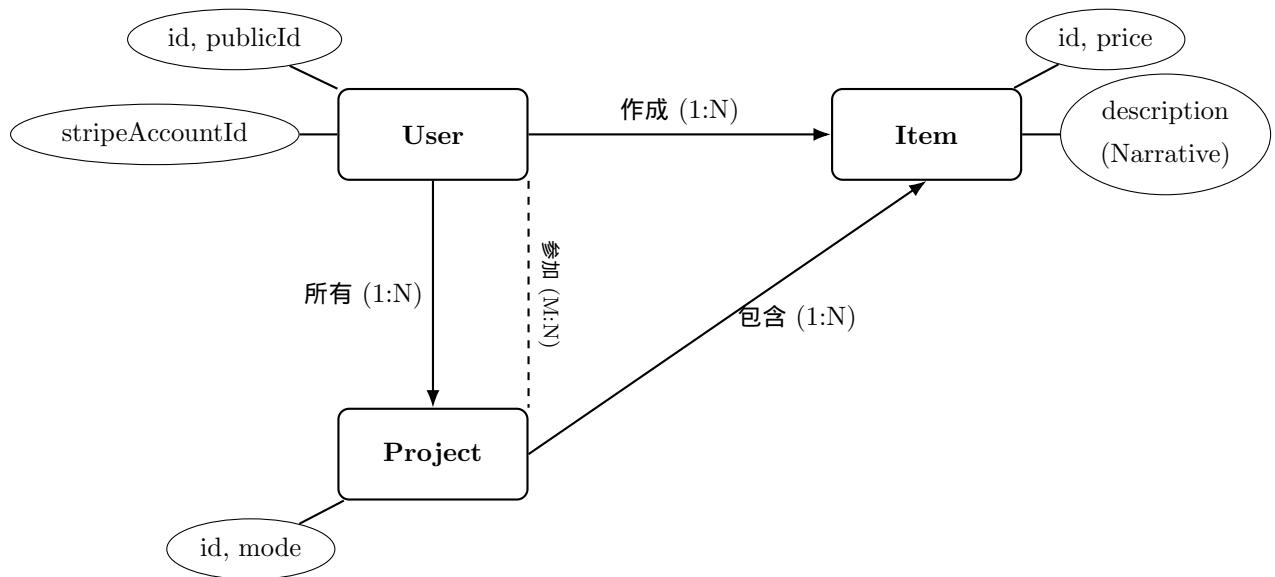


Figure3 User, Item, Project の ER 図 (主要なリレーションのみ抜粋)

は発生しない。

確定と解放 落札や当選が確定した時点で capture を実行し , 売上を確定させる . 一方 , 落選や高値更新が発生した場合は即座に cancel を実行し , 与信枠を解放する .

この仕組みは , 入札や応募の時点で与信を確保することにより , 当選後の未払いの発生を抑制することを目的としている .

4.3 データモデル設計

本システムのデータモデルは , User , Item , Project の 3 つのコアエンティティを中心に設計されている . 主要なエンティティの関係性を図 3 に示す .

4.3.1 User モデル

User モデルは , 本プラットフォームにおける活動の基点となるユーザーを表すリソースである . 単なる認証用アカウント情報に加え , プロフィール情報や決済連携に必要な属性を保持する .

公開識別子とポートフォリオ機能 各ユーザーには , データベース内部の主キー (id) とは別に , 外部公開用の識別子として publicId (CUID) が付与される . これにより , [https://domain/\[userPubId\]](https://domain/[userPubId]) のように公開識別子を用いた URL を生成できる . この URL は , ユーザーの出品作品や参加プロジェクト , 活動履歴を集約したページとして表

示する。

経済的主体としての属性 CtoC 取引における売り手としての機能を果たすため、Stripe Connect のアカウント ID を stripeAccountId カラムに保持する。これは、本人確認が完了したユーザーにのみ発行される ID であり、このフィールドの有無によって、システムはユーザーが「出品可能」か「購入専用」かを判別する。また、買い手としての情報は stripeCustomerId に紐づけられ、クレジットカード情報等の機微な決済情報はすべて Stripe 側の Vault に保存される設計としている。

多層的なリレーション User は他のリソースと多層的な関係を持つ。自身が作成した作品 (items) だけでなく、他者と共同制作した作品 (contributingItems) や、主催する企画 (ownedProjects), 参加者として関わる企画 (contributingProjects) へのリレーションを保持し、個人の活動の広がりをグラフ構造として表現する。

4.3.2 Item モデル

Item モデルは、取引の対象となる「作品」を管理するリソースである。一般的な EC サイトの商品モデルと比較して、本システムでは作品の説明文や制作背景（制作ログ）を保存する点に重点を置く。

説明文の保存 作品の説明文や制作過程を記録するために、description フィールドには Markdown 形式のテキストデータを保存する。フロントエンド側ではリッチテキストとしてレンダリングし、画像や動画の埋め込みを含めて表示する。また、展示会場等で QR コード経由で参照する用途も想定し、作品情報へアクセスできるようにした。

販売ロジックの抽象化 一点物や限定品など、作品の性質に応じた販売形態を選択可能にするため、salesMethod カラム (ENUM 型) を定義している。

AUCTION イングリッシュオークション形式。currentPrice や endTime と連動し、入札を受け付ける。

LOTTERY 抽選販売形式。事前与信を用いた応募ロジックが適用される。

FIXED_PRICE 定額販売形式。購入と同時に決済を確定する。

ステータス管理 作品のライフサイクルを管理するために、status カラム (ENUM 型) を用いる。出品前の DRAFT、販売中の FOR_SALE、取引成立後の SOLD などの状態

を定義し，状態に応じて操作を制御する．また，Stripe の商品マスタと同期するために stripeProductId および stripePriceId を保持する．

4.3.3 Project モデル

Project モデルは，複数の User と Item を束ねる企画単位のリソースである．本システムでは，Project を権限管理と収益分配の単位として扱う．

ガバナンス構造の表現

Owner プロジェクトの作成者であり，管理者権限を持つ．ownerId 外部キーによって User モデルと一対多で結ばれる．

Collaborators 企画に参加するクリエイター群．中間テーブルを介した多対多のリレーションによって定義される．この構造により，招待制の活動や主催者が承認を行う企画など，参加形態の違いを表現できる．

収益分配の単位 Project は紐づけられた作品群 (items) の集合体として機能する．Item モデルは projectId を外部キーとして持ち，どの Project に属するかを示す．このリレーションに基づき，Project 単位での売上集計や主催者手数料の計算を行う．

4.4 Project 管理方式

本プラットフォームにおける Project は，複数ユーザーの共同制作や販売活動を表す単位であり，参加権限や公開フロー，収益分配に関するルールを持つ．現実世界の創作活動には，大学のサークル活動のような水平的な人間関係に基づくものから，コンテストのように主催者が強い権限を持つ垂直的な構造，あるいはインターネット上のミームのように不特定多数が自律的に参加するものまで，多種多様な形態が存在する．これらを単一のシステム仕様でカバーすることは困難であるため，本システムでは Project リソースに対して 3 つの異なるガバナンスマード（管理方式）を実装した．

4.4.1 Cooperative Mode

意義と適用領域 Cooperative Mode は，既知のメンバーによる小規模から中規模の共同制作を想定したモードである．大学のサークル活動，グループ展，特定のクリエイター同士のコラボレーション企画などを対象とする．

仕様詳細

参加権限 「招待制」を採用する。プロジェクトオーナーまたは既存メンバーからの招待リンク（署名付き URL）を受け取ったユーザーのみが、Collaborator として参加できる。

作品管理 参加メンバーは、自身の判断で作品（Item）を Project に紐づけ、即座に公開することができる。オーナーによる事前承認プロセスは存在しない。

収益構造 売上は各作品の制作者に送金し、主催者手数料は設定できない仕様とした。

4.4.2 Managed Mode

意義と適用領域 Managed Mode は、明確な主催者が存在し、品質管理やレギュレーションの遵守が求められる階層型の共同体である。企業主催のコンテスト、有名インフルエンサーによる企画展、アンソロジーの編纂などを想定している。Managed Mode では、主催者に承認や削除等の管理権限を与え、主催者手数料（Organizer Fee）を設定可能とした。

仕様詳細

参加権限 「公募制」を採用する。ユーザーは Project に対して参加申請を行い、オーナーが管理画面でこれを「承認」または「拒否」することでメンバーシップが確定する。

作品管理 出品された作品は一時的に PENDING ステータスとなり、オーナーが内容を確認し承認するまで公開されない。また、オーナーは規約違反の作品を強制的にプロジェクトから除外する権限を持つ。

収益構造 Stripe Connect の "Separate Charges and Transfers" API を活用し、「主催者手数料」の設定を可能にしている。取引成立時、システムは売上総額からプラットフォーム手数料と主催者手数料を自動的に控除し、残額を作品制作者へ送金する。

4.4.3 Theme Mode

意義と適用領域 Theme Mode は、共通のテーマの下で不特定多数が参加可能なモードである。インターネット上のハッシュタグや季節イベント等のように、参加者が広く分散する状況を想定する。本モードでは参加申請や承認プロセスを設けず、参加および公

開を即時に行えるようにした。

仕様詳細

参加権限 「自由参加」を採用する。アカウントを持つ全ユーザーは、許可を得ることなく即座に Project に参加できる。

作品管理 紐づけられた作品は即時公開される。オーナー（発起人）は、通報対応を除き、他者の作品を削除する権限を持たない。

収益構造 主催者手数料の設定は不可である。売上はすべて作品制作者に送金する。

4.4.4 Project 管理方式の比較

各モードにおける権限と金銭的フローの差異を示す。

Table1 Project 管理方式の比較

方式名	参加フロー	作品公開フロー	主催者権限	収益分配 (Organizer Fee)
Cooperative	招待のみ	即時公開	メンバー除名可	不可
Managed	申請 → 承認	投稿 → 承認	作品削除・否認可	可
Theme	自由参加	即時公開	管理不可	不可

4.5 統合型検索システム

本プラットフォームでは、User, Item, Project の 3 つの主要リソースを横断的に検索できる統合検索機能を実装した。ユーザーの検索意図は、特定の作品の購入だけでなく、企画や作家の活動の把握など多様であるため、単一の検索窓から複数のリソースを対象に検索できる構成とした。

4.5.1 設計思想: 探索的検索への対応

従来の E コマースサイトにおける検索システムは、SKU (最小管理単位) ベースの商品検索に特化しており、ユーザーが明確な購買目的を持っていることを前提とすることが多い。しかし、一点物や創作活動を扱う本プラットフォームにおいては、ユーザーは必ずしも具体的な商品名を検索するとは限らない。「折り紙」「ガレージキット」といったジャンル名や、「学園祭」といったイベント名から検索を開始し、そこから関連するクリエイターや作品群へと興味を広げていく探索的検索の行動様式が見られる。したがって、検索システムは単一のリソースリストを返すだけでなく、複数のリソース種別をまとめ

て提示し、関連情報を辿れる構成が望ましい。

4.5.2 アーキテクチャ：並列クエリ実行モデル

統合検索のバックエンド処理においては、異なるデータ構造を持つ複数テーブルに対する検索を効率的に処理するため、並列クエリ実行モデルを採用した。具体的には、ユーザーが入力した単一の検索クエリに対し、バックエンド（NestJS）は以下の処理を非同期かつ並列に実行する。

Project Search プロジェクト名および概要文に対する全文検索またはベクトル検索。

User Search ユーザー名およびプロフィール文に対する検索。

Item Search 作品名、説明文（Markdown）、タグに対する検索。

これらのクエリは `Promise.all` 等を用いて並行してデータベース（PostgreSQL）に発行され、得られた検索結果を単一のレスポンスオブジェクトに統合してクライアントへ返却する。これにより、リソース種別ごとに検索画面を分割する必要を減らす。

4.5.3 インターフェース：階層的情報の統合表示

検索結果の表示画面においては、情報の重要度とユーザーの認知負荷を考慮し、リソースタイプごとに優先順位を設けた階層的表示を採用した。

上部: 関連プロジェクト 検索結果の上部に、クエリに関連性の高い Project を最大 3 件表示する。Project を文脈単位として扱うため、個別作品より上に配置した。

中部: 関連クリエイター 次に、関連する User を最大 5 件表示する。

下部: 関連作品 最下部には、関連する Item をグリッド形式で表示する。ここでは無限スクロールを採用し、追加結果を段階的に取得・表示する。

また、検索結果画面のサイドバーには、属性による絞り込み（ファセットナビゲーション）を配置した。ユーザーは、統合検索によって得られた結果に対し、価格帯、カテゴリ、販売方式（オークション・定額など）といった属性で絞り込みを行うことができる。

4.6 販売形態と価格決定アルゴリズム

本プラットフォームが扱う一点物や希少在庫の流通では、需要が供給を上回る状況が想定される。固定価格販売のみでは先着順となりやすいため、本システムでは複数の販売

方式を実装した。また、全ての方式において、Stripe Connect API を用いたクレジットカードの与信枠確保を利用し、支払い不履行の発生を抑制することを目的とした。

4.6.1 イングリッシュ・オークション

在庫が単一 ($N = 1$) である一点物の作品に対し、最も高い評価額を持つ購入希望者を決定するための標準的な競り上げ方式である。

自動入札アルゴリズム ユーザーの操作量とサーバー負荷の軽減を考慮し、Vickrey Auction の概念を取り入れた自動入札方式を採用した。ユーザーは「現在価格」ではなく、自身の「支払ってもよい最高額」を入力する。システムは、2番目に高い入札額（または開始価格）に最小入札単位を加えた額を「現在価格」として算出・更新する。この方式により、最高額の提示後に再入札を繰り返す操作を減らすことを狙う。

連続的な与信枠管理フロー 本システムでは、最高入札者の入れ替わりに合わせて与信枠を更新するため、最高入札者が変化するたびに決済リソースの切り替えを行う。具体的なフローは以下の通りである。

1. 新規与信 新たな高値入札者が現れた際、システムはそのユーザーのクレジットカードに対し、入札額全額の与信枠確保 (paymentIntents.create with capture_method: 'manual') を試行する。これに失敗した場合、入札 자체を却下する。
2. DB 更新 与信確保に成功した場合のみ、データベース上の Item レコード（現在価格、リーダー ID）および Bid レコードを更新する。
3. 旧与信解放 更新完了後、直前の最高入札者の与信枠 (PaymentIntent) に対してキャンセル処理 (paymentIntents.cancel) を非同期的に実行し、枠を即時解放する。

このプロセスにより、オークション終了時の支払い不履行の発生を抑制する。

4.6.2 複数点式オークション

在庫が複数 ($N > 1$) 存在するが、需要がそれを上回る場合（例：限定 5 個のガレージキット）に適用する販売方式である。当選者の支払額を入札額とする Pay-as-Bid 方式を採用した。

当選者決定ロジック 入札額の高い順にソートを行い，上位 N 名を当選者として確定する．

$$B_1 \geq B_2 \geq \dots \geq B_N \geq \dots \geq B_M$$

ここで B_i は i 番目のユーザーの入札額であり，上位 N 名までの入札が有効となる．同額入札が存在する場合は，先着優先として順位付けを行う．

Pay-as-Bid 方式 当選者全員が B_N (ボーダーライン価格) を支払う単一価格方式ではなく，各当選者が自身の提示した B_i を支払う Pay-as-Bid 方式を採用した．この方式では，当選者ごとに支払額が異なるため，落札結果の支払額は入札額に依存する．

取引および決済フローの詳細 本方式では，複数の当選候補者が並存するため，与信確保と解放を複数件に対して行う．

Step 1: 入札と与信確保 ユーザーが入札を行う際，システムは入札額全額の与信枠確保を実行する．この時点で決済は確定しないが，カードの利用枠を確保することで，入札後の支払い拒否の発生を抑制する．

Step 2: ボーダーラインの可視化と競争 オークション期間中，入札画面には「現在，上位 N 位に入るための最低価格 (B_N)」がリアルタイムで表示される．ユーザーはこのボーダーラインを参考に，自身の予算内で入札額を調整する．

Step 3: 決着と一括決済 決着と一括決済 (Settlement) 終了時刻 (endTime) に到達した時点で，システムは全入札を確定させる．上位 N 名: 確保していた与信枠に対し capture を実行し，売上を確定させる．落選者 ($N + 1$ 位以下): 確保していた与信枠に対し cancel (Void) を実行し，即座に解放する．

4.6.3 抽選販売

価格の高騰の抑制や機会の平等を目的とする場合に用いる方式である．本方式では，応募プロセスにおいて事前与信を行う．

フローとステート管理

応募時 ユーザーが「応募」ボタンを押下した瞬間に，商品価格分の与信枠確保を実行する．確保に失敗した場合，応募は受理されない．

抽選時 擬似乱数生成器を用いて当選者を選出する．

確定時 当選者の取引ステータスを CAPTURED に更新し，決済を確定させる。同時に，落選者に対しては VOID 处理を行い，与信枠を即時解放する。

この仕組みにより，当選後の未払いによる再抽選の発生を抑制し，運用負荷の低減を図る。

4.6.4 定額販売

在庫が無制限，あるいは需給バランスが安定している作品向けの，出展者が価格を決定し販売する E コマースと同様の販売方式である。クレジットカードによる即時決済 (Capture) によって取引が成立する。

4.7 テストデータに対する要件と課題

このように文脈を扱い，かつ複雑なリレーションを持つプラットフォームの開発において，テストデータの用意は課題となる。

1. リレーションの整合性: ランダムなデータを生成すると，「ユーザーが未参加のプロジェクトに関連する作品を出品している」といった外部キー制約違反や論理矛盾が頻発する。これを防ぐためには，データベースの依存関係を深く理解した生成ロジックが必要となる。
2. コンテンツのリアリティ: UI の検証（例えば，長い説明文がレイアウト崩れを起こさないか，検索機能が適切にキーワードを拾うか）を行うためには，「Lorem Ipsum」に代表される無意味なダミーテキストではなく，ドメイン固有の用語や適切な長さを持つ「リアルなテキスト」が必要である。特に本プラットフォームでは「制作ログ」や「物語」が主要なコンテンツであるため，その質が UX 評価に直結する。
3. 多様性の確保: 検索アルゴリズムやレコメンデーション機能の性能を評価するためには，データの分布に偏りがあってはならない。特定のカテゴリ（例：アクセサリー）だけにデータが集中すると，システムの性能を正しく評価できない。

第 3 章で提案したデータ生成エージェントは，これらの課題，特に整合性と多様性の両立を目的として設計した。本プラットフォームは，提案手法の有効性を評価するための実験環境として用いた。

5 評価実験

本章では、提案手法である「RAG を用いたデータ生成エージェント」の有効性を検証するために実施した比較実験の結果について述べる。本実験では、過去の生成履歴を参照する記憶機構（RAG）の導入が、生成される社会的シナリオの多様性と品質に与える影響を評価する。具体的には、RAG により類似内容の重複生成（モード崩壊）の発生が抑制されるか、および生成品質が維持されるかを、定量評価と定性評価の両面から検証する。

5.1 実験設定

以下の条件において、それぞれ 1000 件のシナリオデータ（User-Project-Item のセット）を連続生成し、評価用データセットとした。本実験の目的は、RAG の有無による差を比較可能にすることであり、(1) 語彙的多様性、(2) ベクトル類似度に基づく重複度合い、(3) LLM as a Judge による定性評価、(4) 高類似度、低類似度ペアのシナリオ・実データの定性評価の四つの観点から評価を行った。

Table2 実験条件とパラメータ

実験 ID	基盤モデル (LLM)	RAG (メモリ参照)	データ数
A-1	Gemini 3 Pro	ON	1000
A-2	Gemini 3 Pro	OFF	1000
B-1	Gemini 3 Flash	ON	1000
B-2	Gemini 3 Flash	OFF	1000
C-1	GPT-5.2	ON	1000
C-2	GPT-5.2	OFF	1000
D-1	GPT-5 Mini	ON	1000
D-2	GPT-5 Mini	OFF	1000

すべての条件において、生成力カテゴリは PLASTIC_MODEL, HANDCRAFT, GADGET の 3つを均等に生成するよう指示し、生成の多様性を確保するために Temperature パラメータは 0.9 に設定した。なお、本節でいう生成力カテゴリ（PLASTIC_MODEL 等）は、エージェントの Director ノードで定義されたタスク属性であり、生成するシナリオクラスターがどのジャンルの出品（Item）に対応するかを示すラベルである。Director は、目標分布（targetDistribution）に基づいて次に生成すべきカテゴリを選択し、カテ

ゴリごとに定義された分布マトリクス (DISTRIBUTION_MATRIX) に従って構造タイプ (structureType) をサンプリングし，その組を生成タスクとして下流ノードへ渡す (agent/src/nodes/director.ts).

5.2 生成データの実例

本実験により生成されたデータの一例として，「夜間大学・無線同好会」というテーマに基づいたシナリオをコードリスト 3 に示す．この例では，3 名のユーザー（霧島えみ，矢木拓真，水瀬ナオ）と，彼らが共有するプロジェクト（都市電波ノイズ可視化マップ），および制作された作品（Item）が生成されている．本研究で想定する最小生成単位（User-Project-Item のセット）として，(1) 複数ユーザーが同一 Project に紐づき，(2) 各 Item が作成者（authorId）に対応し，(3) テキスト記述が活動の背景や判断基準を含む，という前提が満たされているかを確認するため，本節で例示する．

```
1  {
2      "runId": "2026-01-31T11-47-56-129Z_gpt-5.2-2025-12-11
3          _rag_on",
4      "generatedAt": "2026-01-31T12:03:11.903Z",
5      "scenario": {
6          "theme": "夜間大学の「電波観測・自作アンテナ同好会」が運営する、都市電波
7              ノイズの“見える化”ガジェット共同制作組織"",
8          "category": "GADGET",
9          "structureType": "ORGANIZATION",
10         "vector": [] , // Google gemini-embedding-001 でシナリオと実
11             データのタイトルをベクトル化したものが入っていた（省略のため削除）
12     },
13     "cluster": {
14         "category": "GADGET",
15         "structureType": "ORGANIZATION",
16         "theme": "夜間大学の「電波観測・自作アンテナ同好会」が運営する、都市電波
17             ノイズの“見える化”ガジェット共同制作組織"",
18         "users": [
19             {
20                 "name": "霧島えみ",
21                 "authorId": "user1"
22             },
23             {
24                 "name": "矢木拓真",
25                 "authorId": "user2"
26             },
27             {
28                 "name": "水瀬ナオ",
29                 "authorId": "user3"
30             }
31         ]
32     }
33 }
```

```

16      "tempId": "u_leader",
17      "publicId": "emi-kirishima-sdr",
18      "name": "霧島えみ",
19      "bio": "夜間大学(情報工学)で働きながら学ぶ社会人学生。昼は雑用、夜
    は電波。"\n\都市のノイズは「見えない公害」になりやすい。だから測って、共有
    して、説明できる形にする。と自作アンテナを“趣味の手触り”的まま社会に接続す
    るのが好きです。nSDR\n\受信・観測中心。送信は免許・技適・法令順守が前提。
    規格外の実験はしません。n",
20      "role": "LEADER",
21  },
22  {
23      "tempId": "u_member_antenna",
24      "publicId": "takuma-yagi-loop",
25      "name": "矢木拓真",
26      "bio": "金属加工屋の夜学通い。アンテナの「曲げ」と「固定」の沼担当。
    "\n\ループ、八木、簡易バラン、マストクランプ...現場で使える形に落とすのが得
    意。測定結果の再現性が上がるとテンションが上がります。n",
27      "role": "MEMBER",
28  },
29  {
30      "tempId": "u_member_firmware",
31      "publicId": "nao-dsp-waterfall",
32      "name": "水瀬ナオ",
33      "bio": "DSP組込み寄りの夜間学生。ノイズの“気配”を/に落とす役。
    UI\n\、平均化、ピークホールド、帯域ごとの統計。数字は冷たいけど、可視化は
    人を動かす。法規とリテラシーを説明する文章も書きます。nFFT",
34      "role": "MEMBER",
35  },
36  {
37      "tempId": "u_member_mapper",
38      "publicId": "junpei-ward-map",
39      "name": "大庭潤平",
40      "bio": "地図と現場の人。夜の街を歩いてログを取る担当。"\n\都市ノイ
    ズは、場所と時間と天候で顔が変わる。だから「誰がいつどこでどう測ったか」を記

```

録して、共同校正（同じ手順で測る）を徹底する派。n" ,

```

41      "role": "MEMBER",
42    },
43  {
44    "tempId": "u_member_parts",
45    "publicId": "mika-case-and-parts",
46    "name": "佐伯ミカ",
47    "bio": ケース設計と部品調達の現実担当。"\n\n3プリント筐体、ネジ・
スペーサ、コネクタの選定、在庫のやりくり。『続けられる観測』のための地味な最
適化が好き。D",
48    "role": "MEMBER",
49  },
50 ],
51 "projects": [
52  {
53    "tempId": "p_noise-map",
54    "publicId": "night-sdr-urban-noise-map",
55    "ownerId": "u_leader",
56    "name": "夜間" : 都市電波ノイズ可視化マップ（共同校正・共同出品）
SDR",
57    "description": "# 概要\夜間大学の「電波観測・自作アンテナ同好
会」が運営する、n都市電波ノイズの“見える化”プロジェクトです。ソフトウェア
無線（*****SDR）と、ループ／ハムなどの自作アンテナを使い、街の中で増え続けるスイッチング電源や*****照明、エレベータ、交通インフラ由来の広帯域ノイズを、LED測って・比べて・説明できる形にします。****\n\私たちがやっている
のは「盗聴」や「妨害」ではなく、n受信（観測）と統計です。観測であっても、公開
するときは誤解を生みやすい。だからこそ、作品ページには法規・電波リテラシーを
必ずセットで載せます。*****\n\n---\n\n# 何を作るのか\n- ノイズ可
視化ガジェット（受信用）****\n - 受信機（市販モジュール）+ 簡易前段フィ
ルタ + 校正済みアンテナSDR\n - スマホ／でスペクトラムとウォーターフォー
ル表示PC\n - 位置情報は任意（匿名化・丸め処理前提）\n - 都市電波ノイズの
可視化マップ****\n - 周波数帯別のノイズフロア（例：VHF／、周辺など）
UHFISM\n - 時刻帯別の変動（夜間大学らしく、夜のデータが厚い）
\n\n---\n\n# 共同校正（コア文化）\n同じ街でも、機材・設定・アンテナ・ケーブル損失で結果が割れる。そこで私たちは、作品を単なる“ガジェット販売”で終わらせず、n共同校正を運用しています。****\n\n- 測定手順テンプレ（サンプル
レート、窓関数、平均化回数、相当の扱い）RBW\n - 参照信号（放送・既知キャリア
など）での相対校正\n - アンテナごとの係数メモ（ループ径、ハム素子長、給電点、
簡易バラン有無）\n - 参加者が同地点で測る「突き合わせ日」\n\n共同出品：校
正が取れた機材・手順は、プロジェクト名義のシリーズとして****に共同出品し、
購入者が同じ手順で追試できるようにします。ArtSquare\n\n---\n\n# 法
規・電波リテラシー（必須表記）\n - 受信は原則自由でも、内容の扱い・公開方法・

```

プライバシーには配慮が必要*****\n- 送信を伴う実験は、免許 / 技適 / 法令の範囲でのみ（本プロジェクトは受信観測中心）****\n- ノイズ源の決めつけはしない（推定は推定として書く）\n\n---\n# このプロジェクトの手触り\夜のキャンパスの屋上、風の音、同軸の重さ、アルミパイプの冷たさ。スペクトラムの底が少し持ち上がるだけで、街の生活が透けて見える。n\n\"見える化\"は、対立のためじゃなく、n会話のためにやります。****",

```

58     "mode": "MANAGED",
59     "memberIds": ["u_leader", "u_member_antenna", "
60         u_member_firmware", "u_member_mapper", "u_member_parts"]
61     ,
62     },
63     {
64         "tempId": "p_antenna-builds",
65         "publicId": "loop-and-yagi-coop-builds",
66         "ownerId": "u_leader",
67         "name": "ループ / ハ木アンテナ共同制作：作って測って係数を残す",
68         "description": "# 目的\の性能を語る前に、だいたいアンテナで決まる。だから、nSDR自作アンテナ（ループ / ハ木）を「作例」で終わらせず、測定と係数までセットにして共有します。*****\n\n---\n# 作るもの
69         可搬型の小型ループ（屋上・公園での観測向け）****\n- 2m/70付近の
70         cm小型ハ木（方位性でノイズの“方向”を読む）****\n- 簡易バラン / チョーク、
マスト固定具、ケーブル取り回し部品\n\n---\n# 共同制作のルール\n- 同じ治具・同じ寸法表から作り、個体差の出るポイント（半田量、クランプ圧、同軸の
引き回し）を記録****\n- 「よく飛ぶ」ではなく、受信でどう見えるか（ノイズフ
ロア、ピークの立ち方、取り回し耐性）をログ化****\n- 法規・リテラシー：送信
前提の表現は避け、受信観測として説明\n\n---\n# 出品方針\パツ一式 /
治具データ / 組立手順 / 係数メモをセットにし、購入者が同じ前提で“見える化”に
参加できる形にします。nPDF",
71         "mode": "COOPERATIVE",
72         "memberIds": ["u_leader", "u_member_antenna", "
73             u_member_parts", "u_member_mapper"],
74         },
75         {
76             "tempId": "p_field-kit",
77             "publicId": "night-walk-sdr-field-kit",
78             "ownerId": "u_leader",
79             "name": "夜歩き観測フィールドキット（静かに測って持ち帰る）",
80             "description": "# コンセプト\夜間大学の帰り道、寄り道して測る。

```

そのためのn静かな観測キットを作ります。****\n\大げさな装備ではなく、バッグに入るサイズで、短時間で設置でき、撤収が早い。都市で測る以上、周囲への配慮が最優先です。n\n\nn---\n\n# 内容\n- 受信モジュール+スマホSDR/接続ライドPC\n- ループアンテナ(折りたたみ) or 小型ロッド+簡易フィルタ\n- 風雜音・ハンドリングノイズを減らす固定具\n- ログテンプレ(時刻、場所は丸め、天候、周辺状況メモ)\n\nn---\n\n# リテラシーの明文化\n- 受信した内容を録音・公開しない(スペクトラム統計のみ)\n- 個人・施設を特定しない形での地図化\n- “原因の断定”をしない(推定は推定として扱う)\n\nn\n---\n\n# ねらい\都市ノイズを、怒りの材料ではなく、n改善の入り口にするための道具として届けます。****",

```

76     "mode": "MANAGED",
77
78     "memberIds": ["u_leader", "u_member_mapper", "
79         u_member_firmware", "u_member_parts"],
80
81     },
82
83 ],
84
85 "items": [
86
87     {
88
89         "tempId": "i_sdr-kit-v1",
90
91         "publicId": "urban-noise-sdr-kit-v1",
92
93         "authorId": "u_member_parts",
94
95         "projectId": "p_field-kit",
96
97         "name": "都市ノイズ可視化"キットSDR v1(受信観測・夜歩き仕様)",
98
99         "description": "# これは何?\夜の街でn都市電波ノイズを“見える化”するための、受信観測向け****スターターキットです。スペクトラム/ウォーターフォールを眺めて終わりではなく、SDRログを持ち帰って比較できるところまでを範囲に入れました。****\n\n> 私たちは送信機を作っているのではなく、街にすでに存在する電波環境(ノイズフロアやスプリアスのように見える成分)を、同じ手順で観測できる形に整えています。n\nn---\n\n# セット内容(現実に組める範囲)\n- 受信モジュール(市販)用マウント一式SDR\n- ケーブル取り回し固定(ハンドリングノイズ低減)USB\n- 簡易フィルタ用の小型ケース(交換しやすいネジ構造)\n- ログテンプレ(Markdown/):周波数範囲、平均化設定、時刻、天候、周辺状況PDF\n\n本体は入手性の都合で「対応機種」を提示し、購入者が選べる形にしています。nSDR\nn\n---\n\n# こだわり(地味だけど効く)
n- ケースと固定:都市の観測は歩きながら立ち止まりながら。ケーブルが揺れるだけでスペクトラムが荒れるので、固定点を増やしました。****\n- 再現性:設定値のスクショではなく、数値で残す。あとから同じ条件で追試できるように。
****\n- 公開前提の匿名化:位置情報は丸め(例:数百****~)を推奨。地点特定の圧を下げる運用を同梱。m\nn\n---\n\n# 法規・電波リテラシー(同梱の注意書きより要約)\n- 本キットは受信観測用途です。送信を伴う用途は、免許・技適・法令の範囲で別途。****\n- 受信した内容(音声等)の共有は行わず、スペクトラム統計として扱う運用を推奨。****\n- ノイズ源の断定はしません。推定を書く場合は根拠と不確かさを併記します。n\nn---\n\n# 共同校正への参加
このキットはプロジェクト「夜歩き観測フィールドキット」の共同校正手順に合わせています。購入者は、同梱テンプレでログを取ることで、プロジェクトのn都市電波ノイズ可視化マップに“同じものさし”で参加できます。****",

```

```

88     },
89     {
90         "tempId": "i_loop-foldable",
91         "publicId": "foldable-loop-antenna-nightwalk",
92         "authorId": "u_member_antenna",
93         "projectId": "p_antenna-builds",
94         "name": "折りたたみ式" 小型ループアンテナ（観測用） 共同校正係数  
メモ付き",
95         "description": "# 概要\\夜の観測で取り回しがよく、設置が早い\\n折  
りたたみ式の小型ループアンテナです。自作アンテナは“作って終わり”になりがち  
なので、これは最初から共同校正（同一手順・同一ログ）に載せる前提で設計しまし  
た。*****\\n\\n---\\n\\n# 構造\\n- ループ径：バッグに入るサイズを優先  
(折りたたみヒンジ+固定ノブ)\\n- 給電部：交換可能なプレート構造（現場で壊し  
ても直しやすい）\\n- 同軸の引き回し：ストレスリリーフを多めに（歩行観測で断  
線しがち）\\n\\n---\\n\\n# “係数メモ”とは\\同じ寸法でも、曲げ半径・締め付け  
トルク・接触抵抗で見え方が変わります。\\nそこで、各個体に以下を添付します。  
\\n\\n- 実測した寸法と重量\\n- 参照帯域での相対レベル確認ログ（観測条件込み）  
\\n- ケーブル長・コネクタ種別\\n\\「絶対値の保証」ではなく、\\n比較のための足  
場です。****\\n\\n---\\n\\n# 法規・リテラシー\\n- 本品は受信観測用途。  
\\n- 送信目的の説明はしません（必要な場合は免許・技適・法令確認が前提）。  
\\n- 観測結果の公開は、地点特定や誤認を避ける運用（丸め、時間帯の幅）を推奨。  
\\n\\n---\\n\\n# 現場メモ\\屋上のフェンス際、街灯の下、コンビニの裏手。ノイ  
ズは場所の“癖”として出ます。折りたたみ機構は、そこに行ける回数を増やすため  
の仕掛けです。\\n",
96     },
97     {
98         "tempId": "i_yagi-uhf-compact",
99         "publicId": "compact-uhf-yagi-visualize",
100        "authorId": "u_member_antenna",
101        "projectId": "p_antenna-builds",
102        "name": "小型"ハムアンテナ（方向でノイズを読む） UHF 組立治具  
データ付き",
103        "description": "# 目的\\都市ノイズは「強い\\n弱い」だけでなく、  
どっちから来るかが重要です。小型ハムの方位性を使って、ノイズの“方向”を読  
み、地図の解像度を上げるための作例です。*****\\n\\n---\\n\\n# 内容\\n- ア  
ルミパイプ素子（切断寸法表付き）\\n- ブーム固定クランプ\\n- 給電部（再現し  
やすい形状）\\n- 組立治具の3データ（位置決めがズレると指向性が崩れるため）  
D\\n\\n---\\n\\n# 使い方（観測）\\n- 同一地点で、一定角度刻みでスペクトラム  
を記録\\n- 目立つキャリアではなく、ノイズフロアの持ち上がりを見比べる  
****\\n- 断定は避け、方向の傾向として共有\\n\\n---\\n\\n# 共同校正\\このハ  
ムは「ループ／ハムアンテナ共同制作」プロジェクトの手順に合わせ、\\n\\n- 角度刻  
み\\n- 平均化設定\\n- 記録テンプレ\\を統一しています。\\n\\n\\n---\\n\\n# 法  
規・電波リテラシー\\n- 受信観測用。送信目的での説明・誘導はしません。
```

\n- 公開時は地点特定に配慮し、施設名などは書かない。 \n\n---\n# 作つて分かること\八木は正直で、ちょっと曲がるとすぐ結果に出ます。だからこそ、街のノイズの“方向性”も、嘘をつきにくい形で残せます。 n",

```

104     },
105     {
106         "tempId": "i_waterfall-app-skin",
107         "publicId": "night-ward-waterfall-ui-pack",
108         "authorId": "u_member_firmware",
109         "projectId": "p_noise-map",
110         "name": "ウォーターフォール可視化"パック（夜間観測用テーマ + ログ出力設定）UI",
111         "description": "# これは何？\nのスペクトラム / ウォーターフォール表示を、夜間の現場観測に合わせて調整したnSDR**テーマ + ログ出力プリセットUIです。**\n\見た目の話に見えるけど、狙いは「比較できるログ」を増やすこと。可視化は派手さよりも、同じものを同じ条件で見られることが重要です。
n\n\n---\n# 内容\n- 夜間向けコントラスト設計（暗所で画面が眩しすぎない）
n- ピークホールド / 平均化のプリセット\n- 周波数帯ごとの注釈テンプレ（“ここは都市ノイズが乗りやすい”など一般論）\n- CSV / のログ出力フォーマット例JSON\n\n---\n# 都市電波ノイズの地図化を前提にこのプリセットは、プロジェクト「都市電波ノイズ可視化マップ」の集計に合わせてあります。
n\n- 帯域の切り方\n- 相当の扱いRBW\n- 時刻の丸め\nを統一し、集計時に「見た目が違うだけの差」を減らします。n\n\n---\n# 法規・電波リテラシー（表記テンプレ同梱）
n- 受信内容そのもの（音声等）ではなく、統計として扱う
n- 地点特定・個人特定につながる記述を避ける
n- 原因断定をしない（推定と根拠の書き方ガイドあり）
n\n---\n# 現場目線\n夜、手袋をしたままでも操作できるボタンサイズ。歩道の端で立ち止まても、n1分でログが取れる導線。そういう地味なところを詰めています。",
112     },
113     {
114         "tempId": "i_calibration-sheet",
115         "publicId": "co-calibration-log-template-sdr",
116         "authorId": "u_member_mapper",
117         "projectId": "p_noise-map",
118         "name": "共同校正ログテンプレ（観測条件・匿名化・再現性のための記録シート）",
119         "description": "# 概要\nプロジェクト全体の背骨になる「共同校正」用の記録テンプレです。ガジェットやアンテナが増えるほど、比較が難しくなる。そこで、先に“書式”を固定します。n\n\n---\n# テンプレに含む項目
n- 機材：機種、サンプルレート、ゲイン、有無SDRAGC
n- アンテナ：種類（ループハムロッド）、寸法、給電、バラン有無
n- ケーブル：長さ、損失の目安、コネクタ
n- 観測：周波数範囲、平均化、窓関数、記録時間
n- 環境：天候、周辺（看板LED架線変電設備など“あるもの”的記録）
n- 位置：公開用は丸め（メッシュ半径指定）
n\n\n---\n# なぜ匿名化が必要？
ノイズの話は、すぐに「犯人探
```

し」になりやすい。私たちはそれを避けます。
 地点特定の圧力を下げる
 推定を断定に変えない
 生活者の文脈で話す（改善の会話につなげる）
 法規・電波リテラシー
 受信観測の範囲で扱う
 受信内容の公開は行わない
 公開は統計・傾向として、誤解されにくい書き方で
 使い方
 このシートは単品でも使えますが、同好会のプロジェクトに参加すると、同じフォーマットでデータが集まるため、都市電波ノイズの可視化マップの精度が上がります。
 ”,

```

120      },
121      {
122          "tempId": "i_rooftop-loop-rig",
123          "publicId": "rooftop-loop-mount-quiet-clamp",
124          "authorId": "u_leader",
125          "projectId": "p_field-kit",
126          "name": "屋上用ループ設置クランプ(振動ノイズを減らす静音マウント)"
127          "",
128          "description": "# 目的\\夜の屋上観測で意外と効くのが、風とフェンス由来の“振動”。アンテナが微妙に揺れると、受信系はすぐに不安定になります。そこで、設置クランプをn静音・減衰寄りに振りました。****\\n\\n---\\n\\n# 特徴\\n- フェンス・手すりに傷を付けにくい当て材\\n- 締め付けトルクが一定になりやすいノブ形状\\n- ループの重心がずれにくいオフセット穴\\n- すぐ撤収できる構造(都市の現場は長居しない)\\n\\n---\\n\\n# 観測と再現性\\n同じ場所で測るとき、設置姿勢が毎回違うと結果が揺れます。n\\nこのマウントは、n\\n- 角度の基準\\n- 高さの基準\\を取りやすくし、共同校正のぶれを減らします。
  n\\n\\n---\\n\\n# 法規・電波リテラシー\\n- 受信観測用途。\\n- 施設や個人を特定する形での公開を避ける(テンプレで運用)\\n\\n---\\n\\n# 現場メモ\\n間大学の屋上は静かで、街はうるさい。その差がウォーターフォールに出たとき、『測る意味』が腹落ちします。クランプはその一歩目の道具です。n",
129      ],
130  },
131 }
```

Listing 3 生成されたシナリオデータの例(夜間大学・無線同好会)

コードリスト3は、実験データ(`data.jsonl`)から抽出した1レコードを整形したものである。レコードは実験メタデータ(`runId, generatedAt`)、シナリオの要約(`scenario`)、および生成されたデータ本体(`cluster`)から構成される。`cluster`内には、ユーザー集合(`users`)および企画・成果物に相当するデータが含まれており、各要素は `tempId` により参照可能である。また、`bio` や `description` には、活動の背景や判断基準(例: 法規・運用上の前提、制作上の意図)が記述されており、社会的シナリオとしての文脈が含まれている。

Scenarist Agent の前提および制約は、コードリスト 2 に示すシステムプロンプトに記述されている。同プロンプトでは、web システムが CtoC の創作・ホビー・コレクションのプラットフォームであること、Item が個人が売買・展示可能な範囲の対象であること、およびカテゴリが PLASTIC_MODEL の場合に「本物の兵器ではない」こと等を前提としている。また、tempId によるリレーション構築、publicId の命名制約、および Markdown 形式の説明文等が制約として与えられている。コードリスト 3 ではカテゴリが GADGET であり、成果物は自作アンテナや可視化 UI 設定等の出品として記述されている。これはプロンプトが定める「個人が売買・展示可能な対象」という前提に沿う。また、各要素に tempId と publicId が付与され、ownerId や authorId、projectId 等を通じた参照関係も含まれていることから、プロンプトが要求するリレーション構築および命名制約が反映されていることが確認できる。さらに、description には Markdown 形式の記述が含まれており、説明文に一定の長さと具体性が付与されている。

5.3 語彙数の増加推移

生成されたデータセットの語彙的多様性を測る指標として、テキストに含まれる「ユニーク単語数」の増加推移を解析した。類似した表現が繰り返される場合、生成数の増加に伴って新出語彙の増加が鈍化する可能性があるため、本節では累積ユニーク数の推移を比較する。

5.3.1 解析手法

各シナリオの theme フィールドを対象に分かち書きを行い、累積ユニーク単語数をカウントした。分かち書きおよび集計は、実験用スクリプト(`agent/scripts/experiment/analysis/vocabulary`)に従い、`Intl.Segmenter` (`locale: ja, granularity: word`) でセグメントに分割した後、`isWordLike` が真となるセグメントを単語として採用した。また、`isWordLike` が偽であっても、文字または数字のみからなるセグメントは単語として採用した。各単語は小文字化した上で集合に追加し、生成順に集合サイズを記録することで、生成ステップごとの累積ユニーク数を算出した。

5.3.2 結果と考察

図 4 に、生成ステップ数に伴う語彙数の増加推移を示す。

実験の結果、1000 件生成時点での総語彙数は以下の通りとなった。

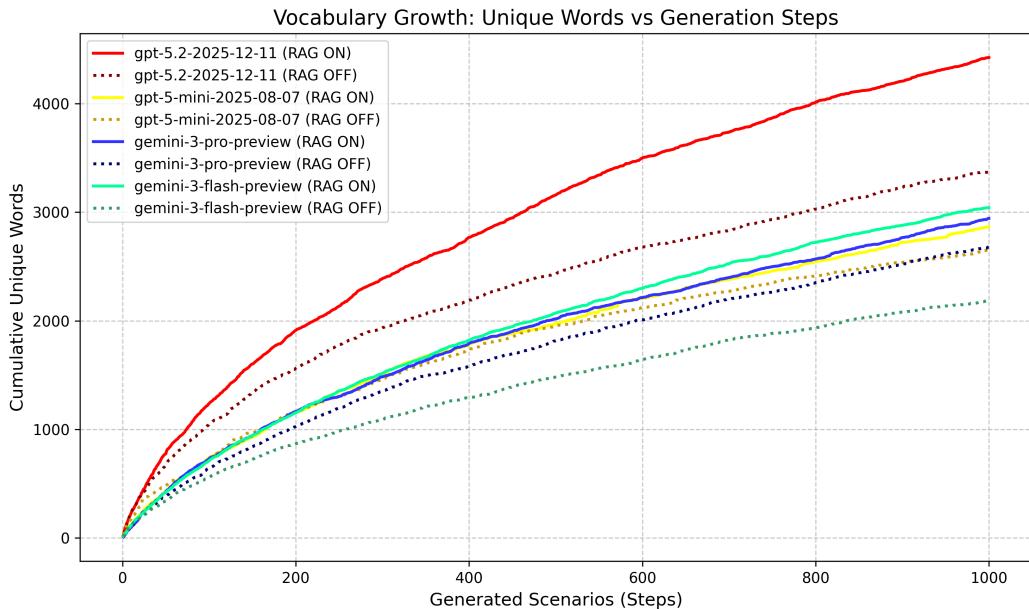


Figure4 生成数に伴うユニーク語彙数の増加推移

- GPT-5.2: RAG ON で 4,425 語に対し , OFF では 3,368 語 (+31.4%)
- Gemini 3 Flash: RAG ON で 3,041 語に対し , OFF では 2,187 語 (+39.0%)

すべてのモデルにおいて , RAG を有効にした場合の方が , より多くの語彙が観測された . 特に増加の推移に注目すると , RAG OFF 群 (図中暖色系) では後半に増加が鈍化する傾向が見られる一方 , RAG ON 群 (図中寒色系) では増加が継続している . この結果は , 本実験の条件下では , 記憶機構により過去の生成物との差別化が促され , 新しい語彙が導入されやすくなる可能性を示唆する .

5.4 ベクトル類似度による重複検知

次に , 生成されたデータ同士がどの程度似通っているかを評価し , RAG の有無によって重複度合いがどのように変化するかを検証した . 本節の目的は , 個別のテーマ文だけでなく , シナリオ全体 (Theme, User, Project, Item) を含む記述の類似性をベクトル空間上で定量化することである .

5.4.1 解析手法

各シナリオに含まれるすべてのエンティティ (Theme, User, Project, Item) のテキストプロパティ (名前 , 説明文 , 属性など) を連結し , 単一の文字列として Embedding モ

モデル (gemini-embedding-001) によりベクトル化した。これにより、単なるあらすじだけでなく、詳細な設定レベルでの類似性を評価した。その後、生成された全ペア ($1000C_2$ 通り) のコサイン類似度分布を計算し、RAG の有無による効果量 (Cohen's d) を算出した。

5.4.2 結果と考察

図 5 にコサイン類似度の分布を示す。また、表 3 に各モデルにおける類似度の変化と効果量を示す。

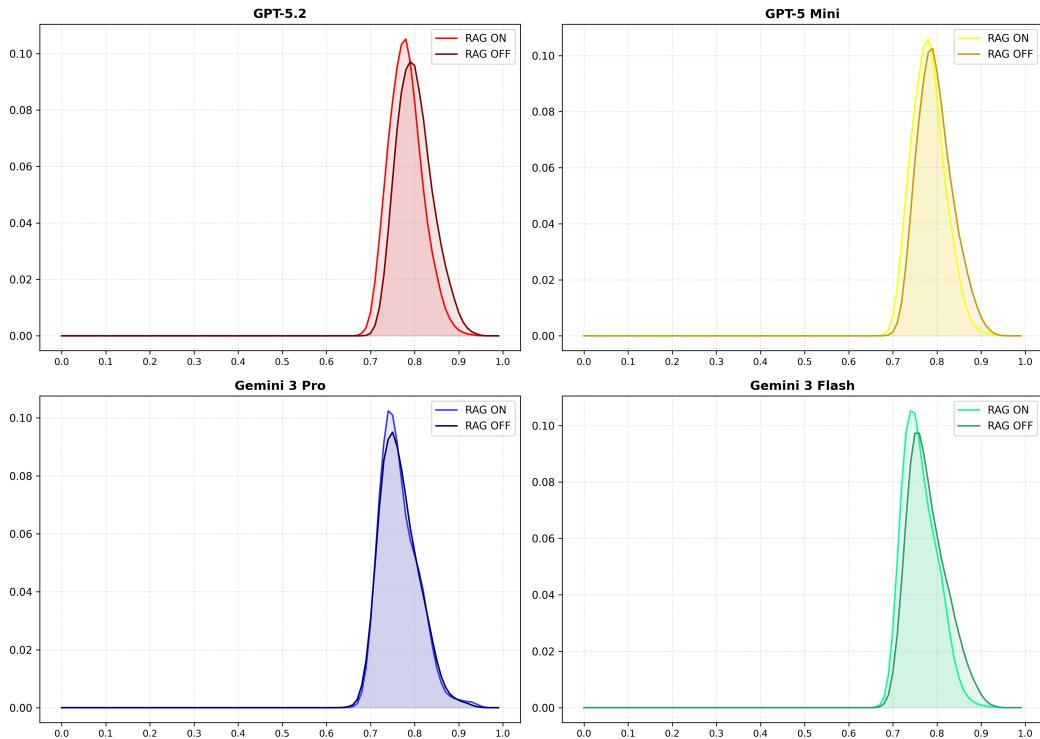


Figure5 生成データ間のコサイン類似度分布 (全プロパティ結合 Embedding)

Table3 RAG 有無による平均類似度の変化と効果量

モデル	平均類似度 (OFF → ON)	差分	Cohen's d
GPT-5.2	0.806 → 0.785	-0.021	-0.51
GPT-5 Mini	0.802 → 0.786	-0.016	-0.40
Gemini 3 Flash	0.783 → 0.767	-0.016	-0.39
Gemini 3 Pro	0.771 → 0.770	-0.001	-0.01

実験の結果、Gemini 3 Pro を除くすべてのモデルにおいて、RAG を有効にすることで類似度の分布が低い側へ移動する傾向が見られた。特に GPT-5.2 においては、効果量

$d = -0.51$ が得られており、RAG ON 条件で平均類似度が低下している（表3）。Gemini 3 Proにおいて差分が小さい点については、本実験の条件下では OFF と ON で平均類似度が近く、RAG による変化が相対的に小さかった可能性がある。

5.5 LLM Judge による定性評価

多様性の向上が品質（整合性や具体性等）の低下を伴わないかを確認するため、Gemini 3 Pro を用いた定性評価を実施した。各条件から 20 件ずつランダムに抽出したシナリオに対し、評価用 LLM が以下の 3 観点で評価を行った（以降、LLM Judge と表記する）。

5.5.1 評価基準

各シナリオに対し、以下の 3 つの観点を 5 段階でスコアリングした。

1. 整合性：設定の矛盾がないか。
2. 具体性：固有名詞、数値、専門用語が含まれているか。
3. 人間らしさ：AI 特有の機械的な記述ではなく、熱量や生活感があるか。

5.5.2 結果概要

全 4 モデルにおける RAG の有無によるスコア比較を表 4 に示す。

Table4 LLM Judge による定性評価結果（各条件 20 件抽出、5 段階評価）

モデル	条件	整合性 (Coherence)	具体性 (Specificity)	人間らしさ (Human-like)
GPT-5 Mini	OFF	5.00	4.75	4.15
	ON	5.00	4.95	4.30
Gemini 3 Flash	OFF	5.00	5.00	4.75
	ON	5.00	5.00	4.95
GPT-5.2	OFF	5.00	5.00	5.00
	ON	5.00	5.00	4.95
Gemini 3 Pro	OFF	5.00	5.00	5.00
	ON	4.95	5.00	5.00

結果として、すべてのモデルにおいて「整合性」は高いスコアを記録した（表4）。また、「具体性」や「人間らしさ」についても、本実験のサンプルでは条件間で大きな差は観測されなかった。

5.5.3 定性的考察：具体性と専門性

生成されたシナリオの内容を確認すると，RAG の有無にかかわらず，専門用語や制作上の判断基準が記述されている事例が見られた。以下に，LLM Judge によって抽出された事例を示す。

事例 1: 専門的具体性の向上

テーマ: 自然風化・天然ウェザリング至上主義「プラスチック盆栽」学会

評価コメント: 「具体的なキット名（RG ザク、サザビー Ver.Ka）設置環境（室外機の上）劣化メカニズム（加水分解、紫外線退色）が極めて詳細かつ論理的に描写されています。塗装 = 欺瞞という独自の哲学にも熱量を感じます。」

事例 2: 独自の視点と人間らしさ

テーマ: 巨大人型兵器の足元にある「都市インフラ」景観保存委員会

評価コメント: 「『ロボット本体には関心がない』という特異なテーマが一貫しており、0.2mm 真鍮線や座屈、応力集中といった専門用語が多用され解像度が高い。AI 特有の不自然を感じさせない、強いこだわりが反映された内容である。」

事例 3: 現代的な風刺とユーモア

テーマ: デジタル世界の『不快な UI』を物理ガジェットとして再現する

評価コメント: 「Cookie 同意バナーや reCAPTCHA といったデジタルストレスを物理化するという発想が秀逸。『トロールの快感』といったブラックユーモアを含む記述には、AI 生成とは思えない強い人間味が感じられます。」

本実験の範囲では，RAG の導入は定性評価スコアに大きな変化を与えない結果となつた。

5.6 数値的類似度と意味的多様性の関係

前節までの評価により，RAG の導入がコサイン類似度を低下させ，語彙数を増加させることが確認された。しかし，Embedding の数値的な類似度が低いことが，必ずしも「人間にとて意味のある多様性」につながっているとは限らない。そこで本節では，GPT-5.2 (RAG ON) および Gemini 3 Pro (RAG ON) において生成されたシナリオペアを類似度のレンジ別に抽出し，その内容を比較することで，数値的な類似度がシナリオの構造や意味の多様性をどのように反映しているかを定性的に検証する。

5.6.1 モデルの違いによる定性的な評価

本節では、高類似度ペアと低類似度ペアを抽出し、テーマや活動目的、登場人物の役割、成果物（Item）の性質、および記述内容の共通点・相違点を比較する。以下に、各モデルにおける事例を示す。

GPT-5.2 における比較 実際の生成テキスト（ユーザー設定および代表アイテム）の比較を図6 および図7 に示す。

シナリオ A: 停電ごっこ運用委員会

User: 黒田イサム（終末論系ラジオ番組の常連投稿者。口は悪いが安全規格にはうるさい「停電ごっこ運用委員会」委員長。週末に「世界が終わった体」で暮らすのが趣味。オフグリッド DIY はロマン、BMS は現実。イベントでは必ずブレーカーを落とす前にチェックリストを読ませるタイプ…）

Item: 停電ごっこスタートー木箱（配電・照明・最低限通信）

Description:

週末の模擬ブラックアウト（停電ごっこ）で、最初に必要なものを「木箱」にまとめたスタートーセット。見た目は終末、中身は現実。

- 5V/12V の簡易配電 / 低消費電力の LED 照明 / 最低限の連絡手段
- 内容物: ヒューズ付き配電ブロック（12V 系）、USB 5V 取り出し（過電流保護あり）、電圧表示（夜間視認優先の控えめ輝度）
- 注意: バッテリーは付属しません（電池は各人の責任、でも事故は全員の責任）。自作バッテリー管理 BMS のチェック項目は同梱の手順書に従ってください。

シナリオ B: ベランダ・マイクログリッド協議会

User: 神名（築14年・400戸規模マンションの管理組合で設備系の議題を担当。ベランダ太陽光と超小型蓄電で“家庭内で完結する省電力ガジェット”を作るのが趣味。DIY は好きだが、共用部ルールと安全監査はもっと大事派。電気工事士ではないので、手を出せる範囲は徹底的に線引きする…）

Item: 窓際ベランダ太陽光 USB 微量充電キット（匿名レビュー付き）

Description:

ベランダ太陽光を“置くだけ”で始めるための、超小型構成の部材キット。マンションの共用部ルールを守るために、手すり外側への張り出しや外壁固定は前提にしていません。

- できること: 晴天の短時間で、USB 機器へ“ちょっと足す”程度の給電。
- 同梱部材: 小型ソーラーパネル、低電流向け充電制御モジュール、小容量セル + 保護基板（構成は監査プロトコルに準拠）、ヒューズ相当（過電流保護）
- 騒音・発火リスク監査: 充電電流は低めに設定。連続充電時の温度ログを添付。

Figure6 GPT-5.2 高類似度ペア (High: 0.89)

高類似度ペア（図6）では、「週末のブラックアウトごっこ」と「マンションでのベランダ発電」という異なるテーマを扱っているが、安全規格や監査プロトコル、配電・保護回路等を重視する点に共通性が見られる。

シナリオ C: コインランドリー常連『回転する時間』

User: ミナ（終電後の街で、コインランドリーだけが明るい。私はそこで回っているものを見ています。洗濯槽の周期、泡の立ち上がり、乾燥の熱が糸に残す癖。服がきれいになる場所というより、都市の生活が一度ほどけて、また巻き取られていく小さな舞台…）

Item: 泡軌道（Foam Orbit）ビアス / イヤリング | 一点物

Description:

ガラス越しに見える泡は、いつも同じようで、同じではない。このビアスは、深夜のコインランドリーで観察した“泡の帯”を、輪郭として固定する試みです。

- 素材: リサイクル繊維（古着コットン）熱で変化する糸。
- 回転パターン採取: 洗濯槽の窓を 12 分割して泡の濃淡を記録。泡が厚く残る区間は編み密度を上げ、途切れが出る区間は糸を飛ばす。
- 同梱ログ: 観察時刻（02:06 ~ 02:33）場所、所感（泡が一度厚くなつてから急に薄くなる）を記した紙片を封入。

シナリオ D: ガンプラ静音研究コミュニティ

User: マキ（平日 22 時以降しか机に座れない社会人モデラー。防音室なし・ワンルーム・隣室あり。「音を出さない工夫 = 制作の自由度」と考えていて、静音工具の比較、サイレント改造の手順、作業音ログを地道に残すタイプ…）

Item: RX 系 クリック関節の“カチ音”低減チューニング

Description:

深夜にボージング確認していると、関節の「カチッ」が響く。保持力を落としきせず、音を丸めるための調整。

- 手順: 分解前に布を敷く（落音対策）。クリック部の当たり面を超微量面取り。受け側に薄いテープを点貼り。
- 作業音ログ: 時間帯（23:40 ~ 00:25）、音の種類（クリック音・高音）、対策（布二重敷き）、体感（「カチッ」「コクリ」と変化）。

Figure7 GPT-5.2 低類似度ペア (Low: 0.74)

一方、低類似度ペア（図7）では、どちらも「ログを取る」という行為を行っているが、対象と目的が異なる。シナリオ C は泡や熱の変化といった観察内容を記録し制作手順に反映しているのに対し、シナリオ D は騒音の指標（例：dB、振動）を記録し静音化を目

的としている。

Gemini 3 Pro における比較 Gemini 3 Pro (RAG ON) の比較事例を図 8 および図 9 に示す。

シナリオ E: AnatomyModeler_K

User: AnatomyModeler_K(架空の『第3工科大学・機械構造学科』で教鞭を執っているという設定のモデル)。模型を「玩具」ではなく「工業製品のサンプル」または「解剖学標本」として捉え、徹底的な内部構造の再現と、それを可視化するためのカットモデル制作を行う…)

Item: 1/100 MS 脚部アクチュエータ構造断面

Description:

汎用人型機動兵器の膝関節周辺を対象としたカットモデル。

- 制作のポイント: 設定画には存在するがキットで省略された「流体バルス伝達パイプ」をスクラッチビルド。
- 切断面の処理: 装甲の切断面は「赤色」で塗装し、教育用・解説用のカットモデルであることを強調(実車カタログ等の技法)。
- 解剖学的視点: 膝装甲の運動スライドギミックと油圧ピストンの同期を観察できるよう、メインアーマーを正中線でカット。

シナリオ F: Dr. Cross-Section / 断面解剖研究所

User: Dr. Cross-Section(「装甲は嘘をつくが、フレームは真実を語る」既成のキットを外科手術の如く切断し、内部メカニズムを露出させる『カットモデル』専攻。ウェザリングはノイズ。清潔な手術室で鋼鉄の巨人の内臓を愛でたい…)

Item: 【RX 系】正中矢状断面: コアブロックシステムの機能的配置

Description:

1/100 スケールキットを使用し、エッティングノコによる手作業で正中線から完全に二分割した作品。

- 解剖学的ポイント: コアブロック周辺の変形ヒンジ構造と、ハイロットシートの衝撃吸收ダンパーをプラス板で新造。
- 塗装: 露出したフレーム断面はシルバー、切断面のエッジを蛍光レッドでハイライトし、「切断面であること」を強調。博物館展示風のフィニッシュ。

Figure8 Gemini 3 Pro 高類似度ペア (High: 0.95)

Gemini 3 Pro の高類似度ペア(図8)は、類似度 0.95 という高い値を示した。両者とも「架空メカの解剖」「切断面の赤色塗装」というポイントが重複している。

シナリオ G: CyberMonk_Logic (サイバー仏教)

User: CyberMonk_Logic(元組み込みエンジニア、現・在宅僧侶。テクノロジーによる功德(Merit)の最大化と自動化を研究する「Cyber-Buddhism」の実践者。はんだ付けは動禅であり、回路設計はマンダラ構築…)

Item: PCB Talisman: 厄除け回路基板 v2.0

Description:

従来の紙製のお守りを、FR-4 ガラスエポキシ基板で再定義した PCB Talisman。

- 技術仕様: 4 層基板の内層に般若心経のバイナリデータを銅箔パターンとして埋め込み、通電によりデジタルな結界を展開。
- 実装: 中央の MCU が乱数生成により 24 時間体制で吉凶をシミュレートし、最適な運気を LED インジケーターで可視化。

シナリオ H: 特定外来生物資源化プロジェクト

User: 郷田博士(元環境保全研究所の生物学者。日本の在来種を脅かす「特定外来生物」の駆除活動に参加する中で、奪った命を廃棄することへの葛藤から、それらを「ジビエレザー」として昇華させる道を選んだ…)

Item: 【特定外来生物】ヌートリアの撥水ファーポーチ

Description:

河川の土手を掘り崩す大型齧歯類「ヌートリア」の毛皮を使用したポーチ。

- 生態学的特徴: 水辺で生活するため、驚異的な撥水性と保温性を持つ。ビーバーに似た濃密な手触り。
- 制作背景: 兵庫県の河川敷で行った個体数調整(駆除)の際に捕獲した個体。肉はコンボスト化し、最も美しい冬毛の部分をなめし加工。命の温かみを道具に変換。

Figure9 Gemini 3 Pro 低類似度ペア (Low: 0.73)

低類似度ペア(図9)では、「デジタルによる精神の自動化(サイバー仏教)」と「狩猟による生命の資源化」という、テクノロジーと自然の対極的な世界観が提示されている。

5.7 まとめ

以上の実験結果より、RAG を用いた記憶機構は、データの重複抑制や多様性の確保に寄与する可能性が示された。

6 結論

本研究では、一点物プラットフォームの開発において、社会的シナリオやそれを反映したスキーマを持つテストデータ生成の課題に対し、RAG を搭載した LLM エージェント

システムを提案した。また、その有効性を検証するための実験環境として、Web プラットフォームを構築した。

従来の手法では再現が困難であった「社会的シナリオ」に基づくりレーションナルデータについて、記憶機構を組み込んだ生成エージェントシステムにより、整合性を保ったデータを自動生成する枠組みを示した。

提案システムの有効性を検証するために実施した比較実験により、以下の知見が得られた。

1. 多様性と重複抑制: 履歴をデータベースに登録し、それを参照しながらデータを生成することで、生成されるテキストの 1000 件時点でのユニーク語彙数が Gemini 3 Flash において 39.0% 増加した。また、生成された社会シナリオおよび各エンティティのテキストをエンベディングし、全ペアのコサイン類似度分布を分析した結果、GPT-5.2 において効果量 $d = -0.51$ が得られた。これらの結果は、本実験条件下では、記憶機構が生成内容の重複抑制と多様性に影響しうることを示唆する。
2. 文脈的一貫性の確認: 定性評価の結果、「夜間大学の無線同好会」の事例に見られるように、User の経歴から Item の技術的仕様に至るまで、文脈に沿った記述が含まれるデータが生成されることが確認された。

今後の課題として、本エージェントの実践的有用性の検証が挙げられる。本研究では、文脈を持つテストデータの生成までを主眼としたが、本来の目的はこのデータを用いてアプリケーションの高度な機能を開発することにある。特に、文脈を考慮した検索システム、複雑なリレーションを可視化する検索 UI、あるいはユーザーの行動文脈に基づいたレコメンデーションシステムといった機能は、意味のあるデータが存在しなければ開発も評価も難しい。今後は、本エージェントによって生成されたシナリオデータを活用し、これらの機能を実際に開発・評価することで、本手法が Web サービス開発プロセスにおいてどのように寄与しうるかを検討する。

さらに、本手法の一般化も重要な課題である。現在は、一点物売買プラットフォームという特定のドメインに特化したデータモデルを用いているが、エージェントのアーキテクチャ自体は汎用的なものである。今後は、この枠組みを他の Web サービスや異なるドメインのアプリケーションにも適用できるよう、前提条件やデータモデル依存を整理し、適用範囲の検討を行う。

謝辞

本研究を遂行するにあたり，多大なるご指導とご助言を賜りました，指導教員の植松幸生准教授に深く感謝いたします．また，日頃より議論を交わし，有益な示唆を頂いた研究室の皆様に心より感謝申し上げます．

参考文献

- [1] Eric Sheridan. *The creator economy could approach half-a-trillion dollars by 2027*. Goldman Sachs Research. Accessed: 2026-02-12. 2023. URL: <https://www.goldmansachs.com/intelligence/pages/the-creator-economy-could-approach-half-a-trillion-dollars-by-2027.html>.
- [2] Inc. Mercari. *Mercari: Your Marketplace*. <https://jp.mercari.com/>. Accessed: 2026-02-12. 2024.
- [3] Inc. GMO Pepabo. *minne — ハンドメイド・手作り・クラフト作品のマーケット*. <https://minne.com/>. Accessed: 2026-02-12. 2024.
- [4] Inc. BASE. *BASE (ベース) — ネットショッピングを無料で簡単に作成*. <https://thebase.com/>. Accessed: 2026-02-12. 2024.
- [5] pixiv Inc. *BOOTH - 創作物の総合マーケット*. <https://booth.pm/ja>. Accessed: 2026-02-12. 2024.
- [6] Marak Squires et al. *Faker.js: generate massive amounts of fake data in the browser and node.js*. <https://github.com/faker-js/faker>. 2022.
- [7] J Edvardsson. “A survey on automatic test data generation”. In: *Proceedings of the Second Conference on Computer Science and Engineering in Linköping* (1999), pp. 21–28.
- [8] DiUS. *JavaFaker*. <https://github.com/DiUS/java-faker>. Accessed: 2026-02-02. 2024.
- [9] Mockaroo. *Random Data Generator and API Mocking Tool*. <https://www.mockaroo.com/>. Accessed: 2026-02-02. 2024.
- [10] Prisma Data. *Prisma: Next-generation ORM for Node.js and TypeScript*. <https://www.prisma.io/>. Accessed: 2026-02-02. 2024.
- [11] Lei Xu et al. “Modeling tabular data using conditional gan”. In: *Advances in Neural Information Processing Systems*. Vol. 32. 2019.
- [12] Martin Jurkovič, Valter Hudoverník, and Erik Štrumbelj. “SyntheRela: A Benchmark For Synthetic Relational Database Generation”. In: *ICLR 2025 Workshop on SynthData*. 2025.
- [13] Haoyang Li et al. “OmniSQL: Synthesizing High-quality Text-to-SQL Data at Scale”. In: *Proceedings of the VLDB Endowment*. 2025.
- [14] Mushtari Sadia et al. “SQuID: Synthesizing Relational Databases from Unstructured Text”. In: *arXiv preprint arXiv:2505.19025* (2025).
- [15] Kyungho Kim et al. “ReFuGe: Feature Generation for Prediction Tasks on Relational Databases with LLM Agents”. In: *Proceedings of the ACM Web Conference 2026*. 2026.
- [16] Joon Sung Park et al. “Generative agents: Interactive simulacra of human behavior”. In: *Proceedings of the 36th Annual ACM Symposium on User Interface Software and Technology*. 2023, pp. 1–22.

- [17] Harrison Chase. *LangChain*. <https://github.com/langchain-ai/langchain>. Accessed: 2026-02-02. 2022.
- [18] Shunyu Yao et al. “ReAct: Synergizing Reasoning and Acting in Language Models”. In: *International Conference on Learning Representations (ICLR)*. 2023.
- [19] Vercel. *Next.js by Vercel - The React Framework*. <https://nextjs.org/>. Accessed: 2026-02-02. 2024.
- [20] Kamil Mysliwiec et al. *NestJS - A progressive Node.js framework*. <https://nestjs.com/>. Accessed: 2026-02-02. 2024.
- [21] PostgreSQL Global Development Group. *PostgreSQL: The World’s Most Advanced Open Source Relational Database*. <https://www.postgresql.org/>. Accessed: 2026-02-02. 2024.
- [22] Stripe. *Stripe API Reference*. <https://stripe.com/docs/api>. Accessed: 2026-02-02. 2024.