

---

# 実ビジネスにおける大規模言語モデルのユースケース探索と実現可能性検討

Exploration of Use Cases and Feasibility Study for Large Language Models in Business

---

中山 佳紀\* 関口 優\* 近藤 宏\* 土山 和華子\* 大向 雅歌\* 篠宮 聖彦\*  
Yoshiki NAKAYAMA Yu SEKIGUCHI Hiroshi KONDOH Wakako TSUCHIYAMA Miyaka OMUKAI Kiyohiko SHINOMIYA

---

## 要 旨

---

近年、GPT-3などの大規模言語モデルが発展し、実際のビジネスにも応用され始めている。それに伴い「プロンプトエンジニアリング」と呼ばれる自然言語で大規模言語モデルを制御する新しい技術分野が誕生している。この技術によって、大規模言語モデルはクリエイティブなアイデアの生成、顧客対応の自動化、意思決定のサポートなど、特定の要件に合わせた応答を生成できるようになり、多方面のビジネス利用が期待されている。

しかし、プロンプトエンジニアリングの方法論や活用事例に関する情報は、様々なソースに分散しており、体系的にまとめられた文献はまだ十分でない。この問題を解決するために、本稿ではプロンプトエンジニアリングに関する知見を体系的に整理し、実際のビジネスにおける大規模言語モデルのユースケースを調査している。また、具体的な活用例として、リコーで検討されている二つのユースケースを通じて、実業務への適応可能性を示している。本稿は、労働者が大規模言語モデルを活用することで生産性を向上させ、より創造的で人間にしかできない業務に集中できるようにし、「はたらく喜び」を高めることを目指している。

## ABSTRACT

---

In recent years, large language models (LLMs) such as GPT-3 have seen significant advancements and are beginning to be applied in actual businesses. Along with this, a new technical domain called prompt engineering, which controls LLMs through natural language, has emerged. With this technology, LLMs are becoming capable of generating creative ideas, automating customer interactions, and supporting decision-making, tailored to specific requirements, and are expected to become widely used in business.

However, information on methodologies and case studies in prompt engineering are scattered across various sources, and there is still a lack of comprehensive literature on the subject. To address this issue, this paper systematically organizes knowledge about prompt engineering and investigates use cases of LLMs in actual business contexts. Furthermore, as concrete examples, we demonstrate the adaptability of LLMs to real business operations through two of RICOH's use cases. Our aim is to enhance productivity by enabling workers to utilize LLMs, allowing them to focus on more creative and uniquely human tasks, thereby fostering joy at work.

---

\* デジタル戦略部 デジタル技術開発センター  
Digital Technology Development Center, Digital Strategy Division

## 1. 背景と目的

近年の人工知能技術の発展は著しく、特に大規模言語モデル（Large Language Model: LLM）の研究が急速に進展している。一般的に、LLMは数十億・数百億以上のパラメータを含むTransformerアーキテクチャをベースにした言語モデルを指し、大量のコーパスで事前学習することによって、人間に近いレベルの文章を生成することができる。その代表例としては、GPT-3<sup>2)</sup>、GPT-4<sup>3)</sup>、PaLM<sup>4)</sup>、LLaMa<sup>5)</sup>などが挙げられる。これらのLLMは、アカデミックな領域に留まらず、OpenAI社の「ChatGPT」、Microsoft社の「新しいBing」、Google社の「Bard」といったチャットボットとして人々に広く認知されている。特に、ChatGPTはリリースから史上最速でアクティブユーザー数1億人を達成した人気の高いチャットボットであり、多くのPlugin機能もサポートすることで、レストラン予約、科学技術計算、PDF読解、作図などの高度な機能を実現させている。

言語モデルの大規模化の背景には、スケーリング則（Scaling Law）<sup>6)</sup>という重要な法則が関係している。この法則は、言語モデルのパラメータ数、データセットのサイズ、計算量が増加するにつれ、損失（LLMの予測と正解の誤差）がべき乗則に従って減少することを示している。つまり、LLMに時間とリソースを投資することで、性能が向上し、これが各企業間での競争を激化させている。

また、LLMは、プロンプトと呼ばれる指示文章を入力することによって、追加の学習データを必要とせずに、様々な自然言語処理（Natural Language Processing: NLP）タスクを実行できることが知られている。言語モデルが大規模化することによって、発現するその能力は文脈内学習（In-Context Learning: ICL）と呼ばれ、NLPの新しいパラダイムとなった。また、プロンプト構造を工夫することによって、LLMの精度は大きく向上することがあり、その技術はプロンプトエンジニアリング（Prompt Engineering）という名前で広く知られている。プロンプトエンジニアリングとLLMを組み合わせるこ

とで、コンピュータプログラミング<sup>7)</sup>、知的労働<sup>8)</sup>、法律<sup>9)</sup>、薬学<sup>10)</sup>などの多種多様な分野でAIの活用が推進されており、OpenAI社の論文<sup>11)</sup>によれば、「アメリカの労働者の80%が、彼らの持つタスクのうち少なくとも10%がLLMの影響を受ける」と推定している。

このようなLLMを含むAIの急速な発展は、日本の産業界にも変化をもたらしている。例えば、リコーでは、数年前から「仕事のAI<sup>12,13)</sup>」という企業固有のドキュメント情報資産を、自然言語処理におけるAI技術によって文章の意味を理解して体系化する事業を立ち上げ、BERT<sup>14)</sup>などの自然言語理解（Natural Language Understanding: NLU）の技術を活用したサービスを展開してきた。分類タスクでは、顧客の声（Voice of Customer: VOC）をマーケティングや商品企画向けに顧客ニーズの観点で分類するサービス<sup>15)</sup>や食品業界向けに食品安全の観点で分類するサービス<sup>16)</sup>を既にリリースしている。また、クラスタリングタスクにおいては、VOCとFAQ（Frequently Asked Questions）から不足しているFAQ候補を可視化することができる機能を開発している。さらには、顧客自身でテキスト分類AIモデルを学習からデプロイまでできるノーコードAI開発ツール、営業日報の分析、提案書のレコメンド、契約書の比較など多岐に渡るドキュメントの業務課題解決に取り組んでいる。近年では、NLUだけでなく、自然言語生成（Natural Language Generation: NLG）の領域における研究開発も盛んに行われている。麻場ら<sup>17)</sup>は、日本語言語資源を用いた60億パラメータサイズのLLMの事前学習及び性能評価を行い、日本語理解ベンチマークJGLUE<sup>18)</sup>の質問応答タスクで、英語のLLMよりも高い性能を示した。

しかしながら、日本の産業界において、LLM学習の技術課題検討が先行する一方で、推論時における重要な技術であるプロンプトエンジニアリングや実ビジネスでのLLM活用ユースケースについて調査を行った文献はまだ十分でない。そこで本稿では、プロンプトエンジニアリングとその応用領域に焦点を当て、プロンプトエンジニアリング初学者やテキ

ストデータを業務で取り扱う労働者に向けて、我々の知見をまとめている。具体的には、下記3点の貢献をもたらしている。

- プロンプトエンジニアリングのノウハウについての体系的な整理
- 実ビジネスにおけるLLM活用アイデアの探索
- アイデアを実現するプロンプトの例示

本稿は、以下のように構成されている。2章では、プロンプトエンジニアリングの基礎や代表的なテクニックを紹介し、さらにその課題についても簡単に触れている。次に、3章では、我々が実施した実ビジネスにおけるユースケース探索の結果を表形式でまとめている。4章では、選定した二つのアイデアに対する具体的な適用例を示しており、5章で本稿を結論づけている。

## 2. プロンプトエンジニアリング

### 2-1 基礎知識

#### 2-1-1 構成要素

LLMにおいて、プロンプトとは、モデルの出力をコントロールするための自然言語で書かれたテキストを指す。そして、プロンプトを効果的に設計し、望む出力を得るための技術やアプローチをプロンプトエンジニアリングと呼ぶ。

Fig. 1に示されているように、プロンプトは以下のような特定の要素から構成されることがある。

- **Instruction (命令)** - LLMが実行するタスクや指示。
- **Context (文脈)** - 背景情報や関連する外部情報。
- **Input Data (入力データ)** - LLMが回答すべき質問やデータ。
- **Output Indicator (出力指示子)** - 出力が開始する場所を示すもの。

上記の内容はSaraviaの「Prompt Engineering Guide<sup>19)</sup>」を参考に記載しており、文献によっては、その定義

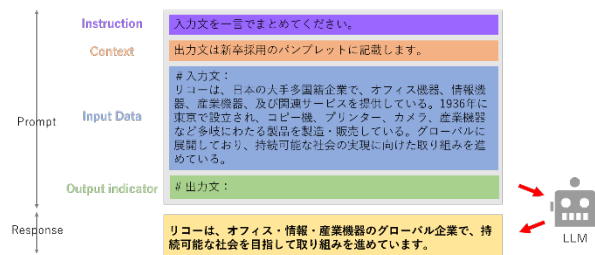


Fig. 1 Example of a prompt for a summarization task.

や構成要素名は異なることがあるため、注意されたい。

他にもRole (役割) やExamples (事例) という要素がプロンプトに含まれる場合もある。Roleを与えるテクニックをRole Promptingと呼び、LLMに医者や弁護士などの特定の役割を付与することができる。また、Examplesを含むプロンプトを作るアプローチはFew-shot Promptingと言い、複数のデモンストレーションをLLMに提示することで回答精度を向上させたり、特定のフォーマットに強制したりすることが可能である。一方、Examplesを与えないアプローチは、Zero-shot Promptingと呼ばれる。Zero-shot Promptingは、Examplesのテキストを試行錯誤する必要がないため、Few-shot Promptingに比べ設計が容易であり、プロンプトのトークン数も節約できる(各LLMには、入力できるトークン数の上限が存在する)。しかし、タスクが複雑であったり、使用するLLMが比較的小規模であったりする場合、期待する精度が得られないことがある。

これらの要素を意識してプロンプトを設計することで、モデルの出力を目的に合わせてコントロールすることができる。

#### 2-1-2 基本的なプロンプト戦略

OpenAI社は、AIの開発やその応用分野での利用において、非常に先駆的である。プロンプトエンジニアリングに関して、以下のベストプラクティス<sup>20)</sup>を提案しており、これらは効果的なプロンプト設計のためのガイドラインとして参考にされている。

- ・ **明確な指示を書く**- 例：詳細な説明をする、ペルソナを設定する、区切り文字を使用する、必要な手順を指定する、具体例を挙げる、出力の長さを指定するなど。
- ・ **参考テキストを提供する**- 例：参考テキストに基づいて回答するようにモデルに指示する、参考テキストの引用部分を答えさせるなど。
- ・ **複雑なタスクをより単純なサブタスクに分割する**- 例：顧客対応においてインテント分類とユーザ支援に分ける、前のターンの会話を要約する、文書を区分的に要約するなど。
- ・ **考える時間を与える**- 例：数学問題の正誤チェックにおいてモデル自身に問題を一度解かせる、前回のパスで見逃しがないかをモデルに尋ねるなど。
- ・ **外部ツールやプログラムコードを利用する**- 例：エンベディングベースの検索を使ってナレッジ検索をする、外部APIでコードを実行させるなど。

さらに、上記ベストプラクティス以外にも、プロンプトエンジニアリングに関するガイドがWeb上に多数存在する。例えば、前述の「Prompt Engineering Guide<sup>19)</sup>」やSchulhoffの「Learn Prompting<sup>21)</sup>」、そしてWengのブログ記事<sup>22)</sup>などがこの分野での知識や技術の習得に役立つと広く認識されている。

### 2-1-3 LLMのハイパーパラメータ

LLMの応答を最適化するためには、プロンプトの設計以外にもハイパーパラメータの調整が重要である。例えば、OpenAI社の提供するAPIでは、Table 1のようなパラメータをサポートしている。これらのハイパーパラメータは、LLMを活用する上で考慮すべき重要な要素である。実際の応用シーンや目的に合わせて、繰り返しテストと調整を行うことで最適な設定を見つける必要がある。

Table 1 Typical hyperparameters of LLM.<sup>23)</sup>

パラメータ名	説明
Temperature	確率分布全体をスケールリングし、出力のランダム性を制御するパラメータ。高い値ではよりランダムな出力になり、低い値ではより確定的な出力になる。
Top P	次トークンの選択肢を調整し、出力のランダム性を制御するためのパラメータ（累積確率の閾値）。核サンプリングとも呼ばれる。高い値ではよりランダムな出力になり、低い値ではより確定的な出力になる。
Max Tokens	生成するトークンの最大数。
Stop (Stop Sequences)	出力停止の目印となるフレーズ。
Frequency Penalty	トークンの出現回数に比例したペナルティ。同じ単語やフレーズを頻繁に繰り返すことを防ぐ。
Presence Penalty	同じトークンが一度でも出現したかどうかに基づくペナルティ。様々な単語やフレーズを出力したい場合に使用される。

## 2-2 応用テクニック

近年、数々の先進的なプロンプトエンジニアリング手法が提案されており、その中でも注目すべき手法を以下で紹介する。

Chain-of-Thought (CoT)<sup>24)</sup> は、Examplesに中間の推論過程を含めることで、入力データに対しても思考過程を明示させる手法である。従来、LLMで算数の文章題のような問題の解決は困難であるとされていたが、CoTを用いることで大幅な精度向上が可能となった。さらに、「Let's think step by step.<sup>25)</sup>」や「Let's work this out in a step by step way to be sure we have the right answer.<sup>26)</sup>」のようなフレーズを組み込んだZero-Shot CoT、Few-Shot CoTを自動で設計するAutoCoT<sup>27)</sup> といった発展形の手法も提案されている。

しかしながら、CoTには一つの推論過程しか取り扱わないという課題がある（人間は通常複数の思考過程を考えることができる）。そこで、Self-Consistency (SC)<sup>28)</sup> では、LLMに同じ質問を複数回投げ、その回答を多数決で選択する方法を取り入れた。SCは前述とCoTと組み合わせることで、算術や常識推論タスクで大幅に精度を向上させた。

また、複数の推論パスを考慮する手法は他にも存在する。Tree of Thought (ToT)<sup>29)</sup>は、LLMが各推論ステップで中間結果を評価する方式を採用し、クロスワードなどのタスクでCoTとSCを組み合わせたアプローチのスコアを上回った。他にもAsk-Me-Anything (AMA)<sup>30)</sup>では、指示文をオープンエンドの質問やyes-no質問に変換する複数のテンプレートを使用し、その変換された質問への回答をもとに、最終的な回答を多数決で決定している。

本節で紹介した応用テクニックは、ごく一部ではあるが、基本的なプロンプトだけで対処できない複雑なタスクや新しいタイプのタスクを解決する際に有効になりうる重要なアプローチである。

## 2-3 プロンプトやLLMの課題

### 2-3-1 事実性

LLMは、抽象型要約や対話生成といった言語生成タスクの性能向上に寄与する一方で、Hallucinationという事実と異なったり、Contextに対して忠実でなかったりする文章を生成してしまう現象が確認されている。我々が知る限りでは、この問題への完全な解決策は、まだ発見されていないが、いくつか緩和策は提案されている。

シンプルなアプローチで言えば、答えが分からない場合に「分かりません」と回答できる旨を明示的にプロンプトに記載する方法<sup>31)</sup>がある。これにより、知識がない事柄に対してモデルが自由な生成をしてしまうことを防ぐことができる。

応用的な方法としては、Retrieval Augmented Generation (RAG)<sup>32)</sup>という手法がある。RAGでは、外部ソースから検索した情報をContextとしてプロンプトに含めることによって、最新の正確な情報に基づいた回答生成をモデルに促すことができ、回答の信頼性を向上させる。

### 2-3-2 プロンプトの脆さ

プロンプトは、その構造のわずかな差異でモデルの出力に大きな影響を与えることが分かっている。

例えば、Zhaoら<sup>33)</sup>は、Examplesの順番や選び方に非常に敏感であることを示した。また、Liuら<sup>34)</sup>は、マルチドキュメント質問応答タスクにおいて、関連情報を位置やContextの長さの違いによる検証を行い、長いContextの中間部分に関連情報がある場合に大きく性能が低下することを発見した。他にも、Pezeshkpourら<sup>35)</sup>の研究では、多肢選択問題において、選択肢の順序に応じてモデルのバイアスが発生することが明らかになっている。

---

## 3. ユースケース探索

---

### 3-1 ユースケース探索の方法

LLMを実際のビジネスへ適用することを考えたとき、自社でLLMを活用するのに最適なユースケースを探索する必要がある。我々は、「活用事例の調査」と「活用アイデアの決定」という二つのフェーズに分けて、ユースケース探索を実施している。

「活用事例の調査」は、様々な情報ソースに掲載されている事例を分野ごとにまとめることで、アイデア発想の元となる知識を把握しやすくすることを目的としている。次の「活用アイデアの決定」では、アイデアの発散と絞り込みを行い、具体的なユースケースを定めていく。絞り込む際は、観点を定め、自社で取り組むべき内容であるか十分に議論していくことが重要である。以降の節では、我々が実際に行った事例調査の結果とリコーにおける具体的なアイデアを記載している。

### 3-2 活用事例の調査

ChatGPTを始めとして、LLMは多くの一般消費者にも知られるようになり、その汎用性の高さを使いやすさから様々なタスクに活用されている。

「GPT-3 DEMO<sup>36)</sup>」や「GPT-4 DEMO<sup>37)</sup>」といった生成AIのアプリケーションカタログサイトには、2023年9月15日時点で計950件以上のアプリケーションが掲載されている。プロンプトエンジニアリング

という観点では、自作したプロンプトを共有するコミュニティサイト<sup>38-40)</sup> やマーケットプレイス<sup>41-43)</sup> も次々に立ち上がっている。他にもX (旧: Twitter) やFacebookなどのSNSでの活用事例の発信も活発である。

これらのソースで紹介されている事例を調査し、その一部をTable 2に分野ごとにまとめた。表に記載した事例は、我々が独自の観点で選定したものであるが、LLMが幅広い分野に応用できることが把握できる。

Table 2 Overview of LLM's practical applications.

分野	活用事例
学会・展示会	FAQ整備, 想定問答集生成, 発表のQ&Aアシスト, トークスクリプト生成
人事・営業	面接・営業練習, 提案書の作成
広報・創造・創作	デザイン案の提示, ブレインストーミング, シナリオ生成, キャッチコピー生成, SEO対策
品質管理・設計	ドキュメント検索, 検索事例の要約
知的労働	進捗管理, タスクリスト生成, スケジューリング, グラフ作成, データ分析, 要因分析, メール作成, 議事録作成, 業務フロー整理
顧客分析	要件定義, 集合人格, ペルソナ作成, ニーズ分析, 課題の掘り下げ
専門的サポート	ITエキスパート, 英会話の練習, 家庭教師, 診療科診断, 知恵袋, 契約書の内容説明, 審査員
多言語理解	通訳, 古代文字の翻訳, 絵文字サジェスション
エンターテインメント	動画コンテンツ要約, 占い, NPC (Non Player Character), クリエイターの分身, 偉人の復活, プロデューサー
ヘルスケア	カウンセリング, 感情類推, 言語化補助, マインドマップ作成, 医療画像診断支援
ライフ	レシピ提案, レストラン予約, 旅行のプランニング
プログラミング	コーディング支援, コードレビュー, Webサイト作成, デバッグ, CLI・Webサーバ・DBのシミュレーション
AI開発	プロンプト生成, アノテーション, 推論根拠の提示, データ生成, クラスタラベリング

### 3-3 活用アイデアの決定

事例調査の結果を参考にし、リコーで取り組む価値があるアイデアについて、関係者間でブレインストーミングを行った。そして、いくつか提案されたアイデアの中から優先度が高いと考えられる五つを選定している。また、選定の際は、以下の三つの観点から検討した。

一つ目の観点は、自社既存サービスとシナジーを持つケースである。リコーでは、2021年に「仕事のAI」事業を開始し、顧客企業の持つ文書データから分類やクラスタリングを含む自然言語理解の機能を提供することに注力してきた。自社のサービスを活用して新しいサービスや機能を提供することは、既存サービスに付加価値をもたらし、さらなる顧客獲得につながると考えられる。

二つ目は、社内データ資源を活用したケースである。リコーは複合機 (Multifunction Peripheral: MFP) やカメラを主力に成長し、現在は幅広い業界へのデジタルソリューションも提供している。長年蓄積されてきたVOCや顧客提案書などの社内データの利活用は、デジタルトランスフォーメーション (Digital Transformation: DX) を推進する上で重要な役割を担い、市場における競合他社との差別化を可能にする。

最後の観点は、自社ドメインに依存しないケースである。前述の二つのケースで自社ドメインを強化していくことは重要である。しかしながら、LLMの潜在能力を一部の用途に限定せず、自社事業と直接的な関係性を持たない領域も戦略的な視点から検討すべきであると考えている。

選ばれた五つのアイデアは、選定理由とともにTable 3にまとめている。選定する際は、対象となる自社サービス、自社データ、技術的ポイントに大きな偏りが生じないように注意した。

まず、自社既存サービスとシナジーを持つケースとしては、「Idea 1: 表データの分析レポート作成」と「Idea 2: 問題点の整理」というアイデアを選んだ。リコーでは、「ノーコードAI開発ツール」という表データを入力としたテキスト分類AIモデル

のノーコード構築サービスを提供している。このサービスに併せて、探索的データ分析（Explanatory Data Analysis: EDA）を簡単に実行できる機能も提供できれば、より顧客価値の高い製品になるため、Idea 1は取り組む価値があると考えられる。さらに、EDAはChatGPTの「Advanced Data Analysis（旧：Code Interpreter）」というLLMがプログラムを実行する機能と非常に相性が良いというも理由の一つである。

Table 3 LLM applications ideas for RICOH.

観点	アイデア	選定理由
自社既存サービスとシナジーを持つケース	Idea 1：表データの分析レポート作成	自社サービス ・ノーコードAI開発ツール 技術的ポイント ・Advanced Data Analysis
	Idea 2：問題点の整理	自社サービス ・VOCクラスタリング 技術的ポイント ・Function Calling
社内データ資源を活用したケース	Idea 3：商材提案資料の作成	自社データ ・提案書 技術的ポイント ・対話型UI
	Idea 4：MFPユーザーとのロールプレイング	自社データ ・MFPについてのVOC 技術的ポイント ・対話型UI
自社ドメインに依存しないケース	Idea 5：食料品注文サイトのレシピアシスタント	想定顧客 ・食品業界 技術的ポイント ・画像生成AIとの連携

また、「仕事のAI」では、VOCを対象にしたクラスタリングサービスにも力を入れている。クラスタリング機能のみでも十分有用であるが、各VOCから顧客が何に対してどう困っているかピックアップできるようなテキスト構造化機能もあると望ましい。このような構造化を実現する技術は、ChatGPTのAPIから「Function Calling」という名前で提供されている。出力形式をJSON形式に強制できるため、APIや呼び出しや情報抽出を実現する手段として重宝されており、Idea 2においても有用だと考えられる。

次に、社内データ資源を活用したケースであるが、こちらは「Idea 3：商材提案資料の作成」と「Idea 4：MFPユーザーとのロールプレイング」の二つを選んだ。Idea 3については、リコーの中の重要な業務である営業活動を支援するアイデアも必要と考えて採用した。

もう一つのIdea 4は、リコーの主力商品であるMFPの販売促進に寄与する可能性の高いアイデアである。MFPの商品企画において、顧客の意見を取り入れることは非常に重要であるが、顧客との実際の対話を任意のタイミングで行うのは難しい。このアイデアが実現することで、顧客の意見を集約し、より顧客志向のMFPを開発することが可能である。

最後の観点である「自社ドメインに依存しないケース」からは、「Idea 5：食料品注文サイトのレシピアシスタント」というアイデアを選んだ。自社ドメイン以外という緩い制約条件のみでは、選定が難しかったため、我々のAI技術が既にサービスインしている食品業界でのニーズを念頭に置き、本アイデアを選定している。さらに、料理はビジュアルが重要な要素であるため、画像生成AIとの統合も考えることができる。

以上が我々の提案するアイデアとその選定理由である。次章では、これらの実現可能性の検討を行うが、誌面の都合上、Idea 1,3に焦点を当て、Idea 2,4,5については割愛する。

## 4. 選定したユースケースにおけるプロンプトエンジニアリングの検証

### 4-1 表データの分析レポート作成

#### 4-1-1 検証の概要

データの特性と構造を深く理解することは、AIモデルを構築する際に非常に重要である。リコーが提供するテキスト分類AIモデルのノーコード構築サービス（ノーコードAI開発ツール）においてもデータを十分に理解した上で利用することでより良

い性能のモデルの構築が可能になると推察される。しかし、統計学やデータ分析の初学者の場合、基本的な統計知識の不足や適切なライブラリや視覚化技術の選択の困難さによって、多くの工数を費やす懸念がある。そこで、リコーの「ノーコードAI開発ツール」への適用を想定し、表データを対象にした分析レポート作成の実現可能性を検証する。

#### 4-1-2 検証環境とモデル

一般的に、LLM単体では統計処理やグラフ描画処理を実行するのは困難である。しかし、2023年9月15日時点で、OpenAI社がChatGPTの有料会員に公開している「Advanced Data Analysis」という機能を使用すると、サンドボックス化された環境の中で、ファイルアップロードからPythonを使った分析、グラフの作成、そしてファイルの保存までの一連の作業が行える。この機能を活用する際は、モデルとしてGPT-4のみが利用可能で、パラメータの調整はできない。我々の知る限り、これが現在最も高精度かつ簡単にデータ分析をLLMで実現できる方法であるため、本アイデアには、Advanced Data Analysis機能をオンにしたGPT-4を使用した。

#### 4-1-3 検証用データ

データセットは、代表的なテキスト分類用日本語データセットであるlivedoorニュースコーパス<sup>44)</sup>を用いた。このデータセットは、NHN Japan社が運営する「livedoorニュース」の中のニュース記事を収集したもので、家電やスポーツに関する9種類のメディア記事から構成されており、合計7,367件のテキストファイルが含まれる。本稿においては、提供元からデータセットをダウンロードし、記事のURL、記事の日付、記事のタイトル、記事の本文、記事の種類といった情報を分割する形でCSVファイルに変換し、ChatGPT上にアップロードした。

#### 4-1-4 プロンプト

プロンプトは、SNS上で広く知られているAdvanced Data Analysis向けのプロンプトフレームワークRASCEF<sup>45)</sup>に従い作成した。RASCEFでは、Role (役割)、Action (行動)、Steps (プロセス)、Context (文脈)、Examples (事例)、Format (出力形式)の要素に分けてプロンプトを構成する。本稿では、データ概要の把握、データ分布のグラフ描画、統計量の算出、分析や洞察といったプロセスを、EDAを行うのに最低限必要なプロセスとし、プロンプトにそれぞれに対応した指示文を含めている。ここでの洞察や分析は、出力したグラフからではなく、算出した統計量から行うことを想定している点に注意されたい(2023年9月15日時点で公開されているGPT-4に画像理解の能力はない)。また、分析レポートのフォーマットについては、マークダウン記法で出力させた。プロンプトの詳細は、付録Aに記載している。

#### 4-1-5 結果

出力されたマークダウンをMarkdown PDF<sup>46)</sup>(マークダウンをPDF化するVisual Studio Code拡張機能)で、PDF化したものをFig. 2に示す。データの概要の把握、基本統計、データ分布のグラフ描画に関して、プログラムを実行して適切な結果を取得していることが確認できた。また、洞察についても事実に基づいた記述がなされていることが確認できた。これは、事前に統計量を算出していることが貢献していると考えられる。

今回使用したAdvanced Data Analysisの機能では、任意のパラメータを設定することはできなかったが、実利用するには、ランダム性を排除したパラメータを設定し、安定した分析結果を提供することが求められる。



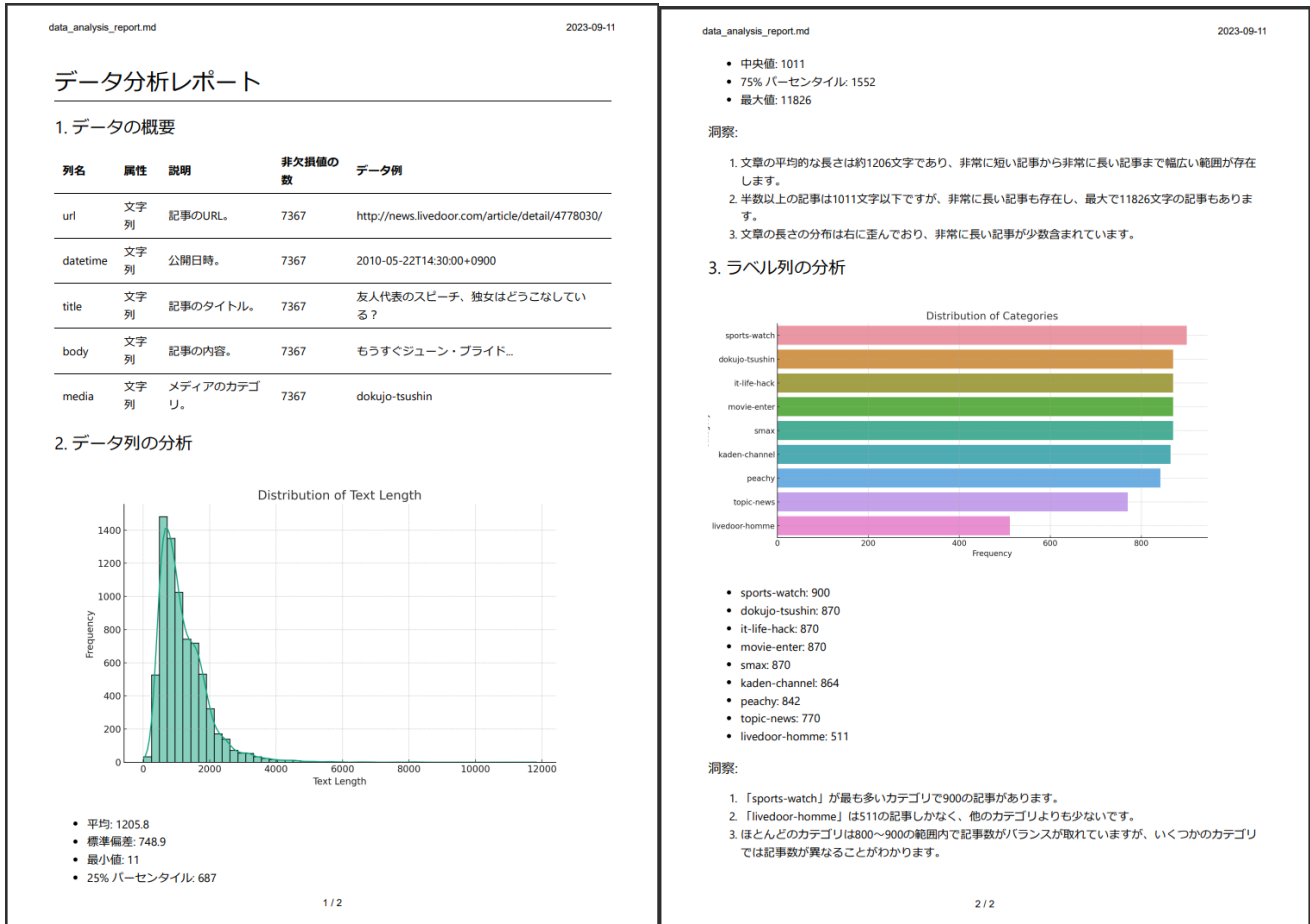


Fig. 2 Data analysis report by GPT-4 with Advanced Data Analysis.

## 4-2 商材提案資料の作成

### 4-2-1 検証の概要

営業のプロセスにおいて、顧客からのヒアリング結果をもとにした提案書作成は、必要不可欠な作業である。この作業の中で、顧客ニーズに対応した商材の選定は必須であるが、新入社員や異動者などの商材に不慣れな担当者が適切な商材を推薦するのは難しい。そこで、本検証では、商談記録と商材情報をもとに、LLMで商材の推薦表が自動生成できるかを検証する。

### 4-2-2 検証環境とモデル

本ケースでは、商談の記録と商材情報をプロンプトに含める必要があるため、比較的トークン数の多いプロンプトにも対応したモデルである gpt-3.5-turbo-16k を用いた。このモデルでは、最大16,385 トークンまでのプロンプトを受け付けることができる。検証はパラメータ変更を容易に行える OpenAI Playground 上で実施した。設定したハイパーパラメータの値は、Table 4 に示している。資料作成タスクにおいて、ランダム性は不要であると考えたため、Temperature の値は 0 にした。また、生成文が途中で打ち切られることを防ぐため、Max Tokens は 8,000 にし、ある程度十分なトークン数を確保している。その他のパラメータに関しては、特に変更を加えずデフォルト値を使用している。

Table 4 Hyperparameter settings for the sales proposal creation in OpenAI Playground.

パラメータ名	設定した値
Temperature	0
Top P	1
Max Tokens	8,000
Stop Sequences	null
Frequency Penalty	0
Presence Penalty	0

#### 4-2-3 検証用データ

本ケースの検証を適切に行うには、リコー社内に蓄積されている過去の商談記録と、それに関連した商材情報の利用が必要である。しかし、OpenAI Playgroundは、外部サービスであるため、業務上の秘密情報をデータとして入力することはできない。そこで、gpt-3.5-turbo-16kを活用し、代替となるダミーデータを生成した。対象となるダミー商材は、建築系のデジタルソリューションとしている。ダミーデータの具体的な内容については、付録Bを参照されたい。

#### 4-2-4 プロンプト

提案書作成タスクは複雑なタスクであるため、いくつかのサブタスク（商談内容の整理、顧客課題の抽出、提案サービスの検討）に分解し、それぞれに対応する指示文を設計した。さらに、LLMの意図しない動作を抑制するため、出力形式やタスク終了時の動作に関する指示もプロンプトに明示的に含めた。

推薦表の出力形式は、マークダウン形式としている。また、ユーザの利便性も考慮し、ユーザ応答の選択肢を提示することで、応答負荷を低減している（追加のプロセスを指示できる選択肢も提供した）。これらのプロンプトに関する詳細な情報は、付録Bに記述している。

#### 4-2-5 結果

LLMが最終的に出力したマークダウン形式の推薦表をMarp for VS Code<sup>47)</sup>（マークダウンをスライド化するVisual Studio Code拡張機能）でスライド資料に変換したものをFig. 3に示す。Fig. 3を見ると、顧客課題に対応した推薦表が作成されていることが分かる。

商材名	解決する問題	導入効果
スピーディ申請ペイペイ	資材や機器の調達方法が分からない	煩雑な申請手続きを効率化し、作業者の時間と労力を節約できる
スマートハウジングアーキテクト	コスト面や効率性を考えて、良いものがあるかもしれないと思っている	効率的で持続可能な住宅の設計に役立ち、コスト削減と効率向上を実現できる

Fig. 3 Product recommendation list by GPT-3.5.

#### 4-3 ユースケース検証から得られた知見

LLMはその高い性能により、不完全なプロンプトを与えられた場合でも一定レベルの成果を提供する能力を備えている。しかしながら、ビジネスの文脈においては、より高度で緻密な成果が要求される。このような要求に対応するためには、プロンプトエンジニアリングにおける細かなテクニックだけではなく、データ分析や提案資料作成などの実業務プロセスの分析と理解をもってプロンプトを設計することが重要であることが検証を通じて示唆された。

また、実際のビジネス環境では、成果物がPDFやスライドのような特定のファイル形式で提供されることが一般的であるが、現時点のLLM単体ではこのレベルの成果物の生成は難しく、外部ツールと連携するのが安定的な出力を得る手段だと考えられる。さらに、本稿での検証は、ChatGPTやOpenAI Playgroundのような対話型の実行環境で行われたが、業務自動化の観点からは、ユーザの入力負荷が大きいという問題が懸念される。例えば、「商材提案資料の作成」のユースケースで示したように、ユーザ入力を選択方式にするなどユーザエクスペリエンスの観点でのプロンプト設計も重要であると推察される。

---

## 5. 結論

---

本稿では、実際のビジネスにおいて、LLMの活用が期待されるユースケースを整理し、プロンプトエンジニアリングに関する一般的なノウハウを共有した。さらに、具体的な例として、「表データの分析レポート作成」「商材提案資料の作成」といったユースケースを取り上げ、実現可能性の検討を行った。結果からは、十分に設計されたプロンプトを用いることで、実業務においてもLLMは人間が作成したかのような高品質な成果物を生成できることが示された。これは、LLMを活用することで、既存業務の効率が改善される可能性があることを示唆している。

さらに、実務に適したプロンプトの設計には、プロンプトエンジニアリングの基礎知識だけではなく、対象業務とそのプロセスに対する深い理解やユーザーエクスペリエンスデザインのスキルも重要であることが判明した。これらのスキルは、LLMを効果的にビジネスに統合し、労働者の「はたらく喜び」を高めることに貢献すると考えられる。

今後の課題としては、より多様なデータに基づく追加の検証、実際の業務で要求される様々なファイルフォーマットへの対応などが挙げられる。

## 参考文献

---

- 1) A. Vaswani et al.: Attention is all you need, *Advances in NeurIPS*, Vol. 30, pp. 5998-6008 (2017).
- 2) T. Brown et al.: Language models are few-shot learners, *Advances in NeurIPS*, Vol. 33, pp. 1877-1901 (2020).
- 3) OpenAI: Gpt-4 technical report, *arXiv preprint arXiv:2303.08774* (2023).
- 4) A. Chowdhery et al.: Palm: Scaling language modeling with pathways, *arXiv preprint arXiv:2204.02311* (2022).
- 5) H. Touvron et al.: Llama: Open and efficient foundation language models, *arXiv preprint arXiv:2302.13971* (2023).
- 6) J. Kaplan et al.: Scaling laws for neural language models, *arXiv preprint arXiv:2001.08361* (2020).
- 7) X. Chen et al.: Teaching large language models to self-debug, *arXiv preprint arXiv:2304.05128* (2023).
- 8) L. Cheng, X. Li, L. Bing: Is gpt-4 a good data analyst?, *arXiv preprint arXiv:2305.15038* (2023).
- 9) J. Savelka et al.: Explaining legal concepts with augmented large language models (gpt-4), *arXiv preprint arXiv:2306.09525* (2023).
- 10) V. Liévin, C. E. Hother, O. Winther: Can large language models reason about medical questions?, *arXiv preprint arXiv:2207.08143* (2022).
- 11) T. Eloundou et al: Gpts are gpts: An early look at the labor market impact potential of large language models, *arXiv preprint arXiv:2303.10130* (2023).
- 12) 鈴木剛, 金崎克己: 「仕事のAI」における大規模言語モデルの個別チューニングを可能にするデータとモデルの共同設計, *Ricoh Technical Report*, No. 45, pp. 38-46 (2023).
- 13) RICOH: 仕事のAI, <https://www.ricoh.co.jp/service/ai-for-work/> (参照2023-09-15).
- 14) J. Devlin et al.: BERT: Pre-training of deep bidirectional transformers for language understanding, *Proc. of NACCL-HLT*, Vol. 1, pp. 4171-4186 (2019).

- 15) RICOH: RICOH ニーズ分析サービス Basic, <https://www.ricoh.co.jp/service/ai-for-work/needs-analysis-service-basic/> (参照2023-09-15).
- 16) RICOH: RICOH 品質分析サービス Standard for 食品業, <https://www.ricoh.co.jp/service/ai-for-work/needs-analysis-service-basic/> (参照2023-09-15).
- 17) 麻場直喜, 梅沢知紀, 川村晋太郎: 日本語に特化した60億パラメータ規模のGPTモデルの構築と評価, 言語処理学会第29回年次大会 (2023).
- 18) 栗原健太郎, 河原大輔, 柴田知秀: JGLUE: 日本語言語理解ベンチマーク, 言語処理学会第28回年次大会, Vol. 29, No. 2, pp. 711-717 (2022).
- 19) E. Saravia: Prompt engineering guide, <https://github.com/dair-ai/Prompt-Engineering-Guide> (参照2023-09-04).
- 20) OpenAI: GPT best practices, <https://platform.openai.com/docs/guides/gpt-best-practices> (参照2023-09-04).
- 21) S. Schulhoff and Community Contributors: Learn prompting, <https://learnprompting.org/> (参照2023-09-04).
- 22) L. Weng: Prompt engineering, <https://lilianweng.github.io/posts/2023-03-15-prompt-engineering/> (参照2023-09-04).
- 23) OpenAI: API reference, <https://platform.openai.com/docs/api-reference> (参照2023-09-04).
- 24) J. Wei et al.: Chain of thought prompting elicits reasoning in large language models, *Advances in NeurIPS* (2022).
- 25) T. Kojima et al.: Large language models are zero-shot reasoners, *Advances in NeurIPS*, Vol. 35, pp. 22199-22213 (2022).
- 26) Y. Zhou et al.: Large language models are human-level prompt engineers, *ICLR* (2023).
- 27) Z. Zhang et al.: Automatic chain of thought prompting in large language models, *ICLR* (2023).
- 28) X. Wang et al.: Self-consistency improves chain of thought reasoning in language models, *ICLR* (2023).
- 29) S. Yao et al.: Tree of Thoughts: Deliberate problem solving with large language models, *arXiv preprint arXiv:2305.10601* (2023).
- 30) S. Arora, et al.: Ask me anything: A simple strategy for prompting language models, *ICLR* (2023).
- 31) ANTHROPIC: Getting started with claude, <https://docs.anthropic.com/claude/docs> (参照2023-09-15).
- 32) P. Lewis et al.: Retrieval-augmented generation for knowledge-intensive NLP tasks, *Advances in NeurIPS*, Vol. 33 (2020).
- 33) Z. Zhao et al.: Calibrate before use: improving few-shot performance of language models, *Proc. of ICML*, Vol. 139, pp. 12697-12706, *PMLR* (2021).
- 34) N. F. Liu et al.: Lost in the middle: how language models use long contexts, *arXiv preprint arXiv:2307.03172* (2023).
- 35) P. Pezeshkpour, E. Hruschka: Large language models sensitivity to the order of options in multiple-choice questions, *arXiv preprint arXiv:2308.11483* (2023).
- 36) GPT-3 DEMO, <https://gpt3demo.com/> (参照2023-09-06).
- 37) GPT-4 DEMO, <https://gpt4demo.com/> (参照2023-09-06).
- 38) Awesome ChatGPT Prompts, <https://www.awesomegptprompts.com/> (参照2023-09-06).
- 39) Snack Prompt, <https://snackprompt.com/> (参照2023-09-06).
- 40) Promptia, <https://prompt.quel.jp/> (参照2023-09-06).
- 41) PromptBase, <https://promptbase.com/> (参照2023-09-06).
- 42) PromptPlus, <https://prompt-plus.ai/> (参照2023-09-06).
- 43) PromptWorks, <https://prompt-works.jp/index.php> (参照2023-09-06).
- 44) RONDHUIT: livedoor ニュースコーパス, <http://www.rondhuit.com/download.html#ldcc> (参照2023-09-11).

- 45) RASCEF, [https://twitter.com/cj\\_zZZz/status/1684630222673698818](https://twitter.com/cj_zZZz/status/1684630222673698818) (参照2023-09-11).
- 46) Markdown PDF, <https://github.com/yzane/vscode-markdown-pdf> (参照2023-09-11).
- 47) Marp for VS Code, <https://github.com/marp-team/marp-vscode> (参照2023-09-11).

- 注1) GPTは、米OpenAI OpCo, LLCの商標です.
- 注2) Bingは、米国Microsoft Corporationの米国およびその他の国における登録商標または商標です.
- 注3) Bardは、Google LLCの商標です.

## 付録

### A. 表データの分析レポート作成に関する実験の詳細

「表データの分析レポート作成」におけるプロンプトの詳細をFig. 4に示す。

<p>Role:</p> <ul style="list-style-type: none"><li>- あなたは優秀なデータサイエンティストです。</li></ul> <p>Action:</p> <ul style="list-style-type: none"><li>- CSV ファイルから最高の EDA を行い、分析レポートを作成してください。</li></ul> <p>Steps:</p> <ol style="list-style-type: none"><li>- 各列について、概要を表にまとめます。</li><li>- データ列について、下記の処理を実行します。<ul style="list-style-type: none"><li>- a) 文字数の分布に関するグラフを保存します。</li><li>- b) 文字数に関する記述統計量を求め、分析と洞察を説明します。</li></ul></li><li>- ラベル列について、下記の処理をします。<ul style="list-style-type: none"><li>- a) カテゴリ別データ数の分布に関するグラフを保存します。</li><li>- b) カテゴリ別データ数を求め、分析と洞察を説明します。</li></ul></li><li>- 結果をすべてレポートにまとめ、そのダウンロードリンクを提供します。</li></ol> <p>Context:</p> <ul style="list-style-type: none"><li>- CSV のデータ列は、「body」です。</li><li>- CSV のラベル列は、「media」です。</li><li>- 分析レポートは、文書分類 AI の構築のために使用されます。</li></ul>	<p>Examples:</p> <p>- Step1: """"</p> <pre>  列名   属性   説明   非欠損値の数   データ例     ---   ---   ---   ---   ---     name   文字列   氏名。   1200   山田太郎     age   数値   年齢。   1200   25  </pre> <p>""""</p> <p>Format:</p> <ul style="list-style-type: none"><li>- 出力はできるだけ簡潔にしてください。</li><li>- 表は、Markdown で記載してください。</li><li>- グラフのタイトルと軸ラベルは、英語にしてください。</li><li>- 分析と洞察は、箇条書きで 3 点程度述べてください。</li><li>- レポートは、できるだけ日本語で書いてください。</li><li>- ダウンロードリンクは、下記をすべて提供し、root は「sandbox」にしてください。<ul style="list-style-type: none"><li>- レポート内容を記載した md ファイル</li><li>- 関連するすべての画像ファイル</li></ul></li></ul>
----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Fig. 4 Prompt for creating tabular data analysis reports.

## B. 商材提案資料の作成に関する実験の詳細

「商材提案資料の作成」におけるプロンプトとその応答履歴の詳細をFig. 5に示す。

<p><b>System:</b></p> <p>あなたは、営業マンを助ける AI です。</p> <p>あなたの目標は、営業マンと対話しながら顧客のヒアリング内容から顧客への提案書を作成します。</p> <p>提案書の内容は、表の形で顧客の問題を解決できる商材をすすめることです。</p> <p>■以下のプロセスを1つずつ実行してください。</p> <p>No.1 まず最初に、前回の商談内容から、顧客の状況を箇条書きで教えてください。</p> <p>No.2 次に、顧客が問題に抱えてそうなことを箇条書きで3点あげてください。</p> <p>No.3 さらに、問題それぞれに対し、提案したい商材の中から問題を解決できそうな商材を選んでください。</p> <p>[問題]</p> <p>[商材]:</p> <p>[理由]:</p> <p>の形で記載してください。理由は2文以内で記載してください。問題にあてはまる商材がない場合は、[商材],[理由]に"なし"と記載してください。</p> <p>No.4 最後に、No.3の解決できそうな商材について、   商材名   解決する問題   導入効果   で表にしてください。</p> <p>すべてのプロセス終了後、営業マンに「お仕事お疲れ様」のねぎらいの言葉をかけて終了です。</p> <p>■プロセス回答形式 (1 ≤ n ≤ 4)</p> <p>No.n</p> <p>&lt;プロセス n の回答&gt;</p> <p>この内容でいかがですか？</p> <p>A:問題ない</p> <p>B:もっと詳しく教えて</p>	<p>C:要望（記載してください）</p> <p>D:1つ前のプロセスに戻る</p> <p>Aなら次のプロセスを実行してください。</p> <p>Bならその回答について具体的に説明してください。</p> <p>Cなら要望に対して修正してください。その場合、プロセス回答形式を無視してください。</p> <p>Dなら1つ前のプロセスの結果を見せてください。</p> <p>###前回の商談</p> <p>営業マン：おはようございます、株式会社〇〇の〇〇でございます。お忙しいところ恐れ入りますが、少しお時間をいただけますでしょうか？</p> <p>顧客：ああ、いいよ。どうしたの？</p> <p>営業マン：ありがとうございます。弊社では建設関連の製品やサービスを提供しておりまして、お客様のニーズにお応えするためにお伺いしました。お客様の現在の建設現場や工事について、詳しく教えていただけますか？</p> <p>顧客：実は新しい建設現場が始まる予定で、そこで必要な資材や機器を探しているんだ。でも、どこで探せばいいのかよく分からなくてね。</p> <p>営業マン：なるほど、新しい建設現場の準備を進められているんですね。弊社では様々な資材や機器を取り扱っておりますので、お手伝いできるかもしれません。具体的に必要な資材や機器についてお教えいただけますか？</p> <p>顧客：特に困っているわけではないけど、コスト面や効率性を考えると、今使っているものよりも良いものがあるかもしれないな。</p> <p>営業マン：おっしゃる通りですね。コスト面や効率性を向上させることは重要です。弊社では数多くの製品やオプションをご用意しておりますので、お客様のニーズに沿ったものをご提案させていただきます。予算の目安やその他の要望はありますか？</p>
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Fig. 5 Dialogue History with GPT-3.5 for sales proposal creation.

顧客：予算の目安はまだ決まっていなくて、コストパフォーマンスの良いものを考えているわ。あと、アフターサービスも大切だと思っているんだけど。

営業マン：了解いたしました。コストパフォーマンスやアフターサービスに重点を置いた製品やサービスをご提案いたします。詳細をヒアリングしてから具体的なご提案をさせていただきたいと思いますが、いつお時間をいただけますでしょうか？

顧客：そうね、まだ詳しい話を進めてから決めたいわ。来週の金曜日の午後なら都合がいいわ。

営業マン：了解いたしました。来週の金曜日、お客様のご都合に合わせてお伺いさせていただきます。事前に提案やデモンストレーションに必要な資料や情報がありましたら、お知らせいただければ幸いです。どうぞよろしくお願いいたします。

顧客：わかりました。こちらこそ、よろしくお願いいたします。金曜日にお会いしましょう。

### ###売りたい商材

#### 1. スマートハウジングアーキテクト: 住宅設計ソフトウェア

- AI 機能を搭載した次世代の住宅設計ソフトウェアです。

- 自動プラン作成やエネルギー効率の最適化など、効率的で持続可能な住宅の設計に役立ちます。

- 豊富なデザインテンプレートとカスタマイズオプションを提供し、クライアントのニーズに合わせたオリジナルな住宅を実現します。

- 直感的なインターフェースと迅速な計算により、効率的な設計プロセスを実現します。

- スマートハウジングアーキテクトは、未来志向の建築プロジェクトにおける革新的なツールとして活躍します。

#### 2. ドリームホームデザイナー: 住宅設計ソフトウェア

- 簡単な操作で美しい住宅デザインを作成できる 3D 設計ソフトウェアです。

- 豊富なライブラリとプリセットモデルを備えてお

り、理想の住宅を直感的にプランニングできます。

- リアルタイムのレンダリング機能により、臨場感のある仮想空間で住宅を体験することができます。

- ドリームホームデザイナーは、建築家やデザイナーだけでなく、一般のユーザーにも使いやすいソフトウェアです。

- クリエイティブなアイデアを視覚化し、理想の住宅を実現するための優れたツールです。

#### 3. ホームワンダーランド: 住宅設計ソフトウェア

- 子供向けの楽しいインターフェースを備えた、住宅設計ソフトウェアです。

- カートゥーン風のグラフィックと対話的な要素が、子供たちのクリエイティブな発想を刺激します。

- シンプルなプリセットツールや簡単な操作方法で、子供たちは自分の夢の家をデザインし、楽しむことができます。

- ホームワンダーランドは、建築やデザインへの興味を持つ子供たちに、創造力と設計スキルを育む素晴らしいツールとなります。

- 家族で楽しく使えるソフトウェアとして、共有の家づくりの楽しさを提供します。

#### 4.スピーディ申請ペイペイ: オンラインビルディング申請システム

- オンライン申請をスムーズに行い、手続きをペイペイ（簡単）なものにするサービスです。

- ユーザーフレンドリーなインターフェースとワンクリック操作により、煩雑な申請手続きを効率化します。

- 豊富な申請テンプレートや自動入力機能を備え、申請書類の作成時間を大幅に短縮します。

- 迅速な審査状況の追跡や通知機能により、申請者は常に最新の情報を把握できます。

- スピーディ申請ペイペイは、建設業界における煩雑な申請プロセスをスムーズにし、作業者の時間と労力を節約します。

Fig. 5 – Continued from previous page.



まず、プロセス 1 からお願いします。

**User:**

No.1

前回の商談内容から、顧客の状況を箇条書きで教えてください。

- 顧客は新しい建設現場の準備を進めている
- 資材や機器の調達方法について分からない
- コスト面や効率性を考えて、良いものがあるかもしれないと思っている
- 予算の目安はまだ決まっていないが、コストパフォーマンスの良いものを考えている
- アフターサービスも重要だと考えている
- 詳しい話を進めてから決めたい
- 来週の金曜日の午後に再度会う予定

この内容でいかがですか？

A:問題ない

B:もっと詳しく教えて

C:要望（記載してください）

D:1つ前のプロセスに戻る

**Assistant:**

問題ないです

No.2

次に、顧客が問題に抱えてそんなことを箇条書きで3点あげてください。

- 資材や機器の調達方法が分からない
- コスト面や効率性を考えて、良いものがあるかもしれないと思っている
- アフターサービスも重要だと考えている

この内容でいかがですか？

A:問題ない

B:もっと詳しく教えて

C:要望（記載してください）

D:1つ前のプロセスに戻る

**User:**

問題ないです

**Assistant:**

No.3

さらに、問題それぞれに対し、提案したい商材の中から

ら問題を解決できそうな商材を選んでください。

[問題] 資材や機器の調達方法が分からない

[商材]: スピーディ申請ペイペイ

[理由]: オンラインビルディング申請システムであり、煩雑な申請手続きを効率化し、作業者の時間と労力を節約できます。

[問題] コスト面や効率性を考えて、良いものがあるかもしれないと思っている

[商材]: スマートハウジングアーキテクト

[理由]: AI 機能を搭載した次世代の住宅設計ソフトウェアであり、自動プラン作成やエネルギー効率の最適化など、効率的で持続可能な住宅の設計に役立ちます。

[問題] アフターサービスも重要だと考えている

[商材]: なし

[理由]: アフターサービスに特化した商材が現在の提案リストにはありません。

この内容でいかがですか？

A:問題ない

B:もっと詳しく教えて

C:要望（記載してください）

D:1つ前のプロセスに戻る

**User:**

いいですね、問題ないです

**Assistant:**

No.4

最後に、No.3 の解決できそうな商材について、|商材名|解決する問題|導入効果|で表にしてください。

|商材名|解決する問題|導入効果|

|---|---|---|

|スピーディ申請ペイペイ|資材や機器の調達方法が分からない|煩雑な申請手続きを効率化し、作業者の時間と労力を節約できる|

|スマートハウジングアーキテクト|コスト面や効率性を考えて、良いものがあるかもしれないと思っている|効率的で持続可能な住宅の設計に役立ち、コスト削減と効率向上を実現できる|

|なし|アフターサービスも重要だと考えている|現在の提案リストにはアフターサービスに特化した商材があ

Fig. 5 – Continued from previous page.

りません|  
この内容でいかがですか？  
A:問題ない  
B:もっと詳しく教えて  
C:要望（記載してください）  
D:1つ前のプロセスに戻る  
**User:**  
なし の部分をなくしてください  
**Assistant:**  
了解しました。以下の表に修正いたします。  
商材名	解決する問題	導入効果
スピーディ申請ペイペイ	資材や機器の調達方法が分からない	煩雑な申請手続きを効率化し、作業者の時間と労力を節約できる
スマートハウジングアーキテクト	コスト面や効率性	

を考えて、良いものがあるかもしれないと思っている  
|効率的で持続可能な住宅の設計に役立ち、コスト削減と効率向上を実現できる|  
この内容でいかがですか？  
A:問題ない  
B:もっと詳しく教えて  
C:要望（記載してください）  
D:1つ前のプロセスに戻る  
**User:**  
問題ないです！  
**Assistant:**  
お仕事お疲れ様でした！営業マンさん、顧客の問題に対して解決できそうな商材を提案するための情報が整いました。次回の商談で、顧客に具体的な提案を行ってください。頑張ってくださいね！

Fig. 5 – Continued from previous page.