

プロンプトエンジニアリング

基本から応用まで

AI初心者・非エンジニアのための実践ガイド

作成日：2025年05月26日

バージョン：Ver. 1.0

作成者：TSUNEHIO TAKEUCHI (AI アシスタント協働)

はじめに

近年、ChatGPTをはじめとする対話型AI（人工知能）が急速に普及し、私たちの仕事や日常生活に大きな変化をもたらしつつあります。これらのAIは、まるで人間と会話するように自然な言葉でやり取りができ、文章作成、アイデア出し、情報収集など、多岐にわたるタスクをこなす能力を持っています。しかし、AIの力を最大限に引き出すためには、私たちユーザーがAIに対して「どのように指示を出すか」が非常に重要になります。この「上手な指示の出し方」こそが、「プロンプトエンジニアリング」と呼ばれる技術です。

本資料の目的

本資料は、AI、特に大規模言語モデル（LLM）と呼ばれる種類のAIと、より効果的にコミュニケーションを取るための「プロンプトエンジニアリング」の基本的な考え方とテクニックを、専門知識がない方にも分かりやすく解説することを目的としています。

- プロンプトエンジニアリングの基本的な概念と、その重要性について理解を深める。
- AI（特に大規模言語モデル）に対して、より効果的な指示（プロンプト）を作成するための基礎知識とテクニックを習得する。
- 日常業務や個人の活動において、AIを有効活用するための第一歩を踏み出す。

対象読者

本資料は、以下のような方々を対象としています。

- AI技術やプロンプトエンジニアリングに初めて触れる方。
- プログラミングの専門知識がない非エンジニアの方。
- ChatGPTなどの対話型AIを使い始めたが、もっと良い回答を得る方法を知りたい方。
- AIを使って、日々の作業を効率化したり、新しいアイデアを生み出したりしたいと考えている方。

本資料の構成

本資料は、プロンプトエンジニアリングの定義から始まり、その背景にある基本概念、具体的な作成テクニック、そして簡単な応用例までを段階的に解説します。専門的な知識がなくても理解できるよう、平易な言葉と具体例を多く用いています。各章の概要は、次の目次に示します。

前提知識

特別な前提知識は必要ありません。基本的なPC操作（キーボード入力、マウス操作）およびインターネットの利用経験があれば十分にご理解いただけます。「AIって何だか難しそう…」と感じている方でも、安心して読み進めてください。

目次

はじめに

- 本資料の目的
- 対象読者
- 本資料の構成
- 前提知識

プロンプトエンジニアリングとは？

- プロンプトの定義：AIとのコミュニケーションの「カギ」
- プロンプトエンジニアリングの重要性：AIの力を最大限に引き出すために
- プロンプトエンジニアリングでできること：広がる活用の可能性

基本概念と理論的背景：プロンプトが機能する「舞台裏」

- 基盤となるAI技術：大規模言語モデル (LLM) ってなんだろう？
- プロンプトの基本構成要素：AIへの指示を分解してみよう
- 指示の明確性がもたらす効果：なぜ「具体的に」が大切なのか？

プロンプト作成の基本テクニック：AIに「伝わる」指示のコツ

- 具体的な指示を出すコツ：「察して」ではなく「伝えて」
- 効果的なコンテキスト（背景情報）の与え方：AIにも「状況説明」が必要
- 役割（ロール）を指定する：AIを「〇〇の専門家」に任命しよう
- 望ましい出力形式を指定する：欲しい答えの「形」を伝えよう

プロンプトエンジニアリングの応用テクニック入門：一歩進んだ使い方

- Few-shot プロンプティング：AIに「お手本」を見せる
- Chain-of-Thought (CoT) プロンプティング：「考え方」をAIに示す
- Zero-shot プロンプティングとの使い分け：状況に応じたテクニック選択

実践演習：AI初心者・非エンジニア向けケーススタディ

- ケーススタディ1：日々の献立のアイデア出し
- ケーススタディ2：ビジネスメールの件名作成

まとめと次のステップ

- 本資料の重要ポイント振り返り
- プロンプトエンジニアリング活用のヒント
- さらなる学習のために

プロンプトエンジニアリングとは？

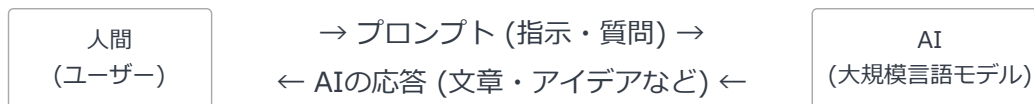
この章では、プロンプトエンジニアリングの世界への第一歩として、まず「プロンプト」そのものが何であるか、そしてなぜ「プロンプトエンジニアリング」という考え方が重要なのか、さらにそれによって何ができるのか、基本的な事柄を解説します。

プロンプトの定義：AIとのコミュニケーションの「カギ」

「プロンプト（Prompt）」とは、とても簡単に言うと、**AI（特にChatGPTのような大規模言語モデル）に対して、私たちが入力する「指示」や「質問」のこと**です。AIに何かをしてほしい時、あるいは何かについて教えてほしい時に、私たちがAIに語りかける「言葉」や「文章」がプロンプトにあたります。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#), Introduction)」によれば、プロンプトは「モデルが特定の出力を予測するために使用する入力」と定義されています。つまり、AIが望ましい答えを出すための「きっかけ」や「手がかり」となる情報なのです。

例えば、あなたがAIに「今日の天気は？」と尋ねるなら、この「今日の天気は？」という言葉全体がプロンプトです。AIはこのプロンプトを解釈し、天気に関する情報を生成しようとします。このように、プロンプトは人間とAIが対話し、協力して何かを生み出すための、まさに「コミュニケーションのカギ」と言えるでしょう。

図1：プロンプトの概念図



説明：人間はAIに対して「プロンプト」という形で指示や質問を投げかけます。AIはプロンプトを解釈し、それに応じた応答を生成して人間に返します。プロンプトは、人間とAIの間のコミュニケーションを仲立ちする重要な「カギ」となります。

プロンプトエンジニアリングの重要性：AIの力を最大限に引き出すために

「プロンプトエンジニアリング（Prompt Engineering）」とは、**AIから期待通りの、あるいは期待以上の質の高い出力を得るために、効果的なプロンプトを設計し、試行錯誤しながら最適化していく技術やプロセス**のことです。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#), Prompt engineering)」では、「LLMが正しいトークン列を予測できるように設定する試み」であり、「高品質なプロンプトを設計するプロセス」と述べられています。

現在のAIは非常に高性能で多機能ですが、まるで魔法のランプの精霊のように、呼び出し方（＝プロンプトの与え方）次第で、その働きぶりは大きく変わります。曖昧な指示や分かりにくいプロンプトでは、AIも混乱してしまい、的外れな答えを返してきたり、意図しない情報を生成したりすることがあります。同資料の「Introduction」セクションでも指摘されているように、不適切なプロンプトは曖昧で不正確な応答に繋がる可能性があります。

だからこそ、AIの能力を最大限に引き出し、私たちが本当に欲しい情報や助けを得るためには、プロンプトを工夫する「プロンプトエンジニアリング」の考え方が不可欠なのです。これは、AIという強力な道具を上手に使いこなすための「取扱説明書」を自分で作り上げていく作業に似ています。

プロンプトエンジニアリングでできること：広がる活用の可能性

プロンプトエンジニアリングを身につけることで、AIを使って様々なことができるようになります。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#), Prompt engineering)」では、以下のようなタスクが挙げられています。

- 文章の要約（長い記事や報告書の内容を短くまとめる）
- 情報抽出（特定の文書から必要な情報だけを取り出す）
- 質疑応答（質問に対して的確な答えを得る）
- テキスト分類（文章をカテゴリー分けする）
- 翻訳（ある言語の文章を別の言語に訳す）
- コード生成（プログラミングのコードを書いてもらう）
- コードの説明（既存のプログラムが何をしているか解説してもらう）

これらは専門的に聞こえるかもしれませんが、非エンジニアの方にとっても、プロンプトエンジニアリングは日々の業務や個人の活動を豊かにする多くの可能性を秘めています。例えば、以下のような活用が考えられます。

- **文章作成支援**：ブログ記事のアイデア出し、メールの文面作成、会議の議事録の要点整理、企画書の構成案作成など。
- **情報収集・整理**：特定のトピックに関する情報収集と要約、複雑なニュース記事のポイント抽出、専門用語の分かりやすい解説など。
- **アイデア発想**：新商品やサービスのキャッチコピー案、イベントの企画アイデア、趣味の創作活動のヒントなど。
- **学習支援**：難解な専門分野の入門的な説明、歴史上の出来事や人物に関する概要整理、外国語の練習相手など。

このように、プロンプトエンジニアリングは、AIとの対話をより建設的で生産的なものにし、私たちの創造性や問題解決能力を高めるための強力なスキルとなるのです。

基本概念と理論的背景：プロンプトが機能する「舞台裏」

プロンプトエンジニアリングを効果的に行うためには、AI、特に大規模言語モデル（LLM）がどのようにして私たちの言葉を理解し、応答を生成しているのか、その基本的な仕組みを知っておくと役立ちます。この章では、プロンプトが機能する「舞台裏」を少し覗いてみましょう。

基盤となるAI技術：大規模言語モデル（LLM）ってなんだろう？

プロンプトエンジニアリングの対象となるAIの多くは、「大規模言語モデル（Large Language Model、略してLLM）」と呼ばれる種類のものです。LLMとは、**非常に大量のテキストデータ（例えば、インターネット上のウェブサイト、書籍、記事など）を学習することで、人間が使う言葉のパターンや文法、単語同士の関連性などを理解し、人間が書いたかのように自然な文章を生成したり、質問に答えたり、翻訳したりする能力を持つAI**の一種です。

参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#), Prompt engineering)」によると、LLMは本質的に「予測エンジン」です。入力されたテキスト（プロンプト）に続いて、次にくる可能性が最も高い単語（より正確には「トークン」と呼ばれる単位）は何かを予測します。そして、その予測された単語を元のテキストに付け加え、さらに次の単語を予測する…という処理を繰り返すことで、一連の文章を生成していくのです。例えば、「昔々あるところに、おじいさんと…」という文章が与えられたら、LLMは次に「おばあさん」という単語が来る確率が高いと予測するでしょう。

LLMは、学習した膨大なデータの中から、言葉と言葉のつながりの強さや、文脈に応じた適切な表現を統計的に判断しています。魔法のように言葉を生み出しているわけではなく、あくまで確率に基づいた予測を行っているのです。

イラスト1：LLMの働き方のイメージ

（ここに、大きな「脳」のようなアイコンがあり、その脳にたくさんの「本」「新聞記事」「ウェブサイトのページ」といった情報が流れ込んでいる様子を想像してください。そして、その脳から、人間が理解できる「言葉」や「文章」が吹き出しのようにアウトプットとして出てくるイメージです。）

LLMは、まるでたくさんの本を読んで勉強した賢いAIです。大量の文章データから言葉の使い方を学習し、私たちが話しかけると、その文脈に最も合う言葉を予測して、人間のように自然な文章で答えてくれます。

プロンプトの基本構成要素：AIへの指示を分解してみよう

効果的なプロンプトを作成するためには、どのような要素を含めると良いのでしょうか？プロンプトは自由な文章で書けますが、AIに意図を正確に伝えるためには、いくつかの基本的な構成要素を意識すると役立ちます。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」のベストプラクティスやプロンプト例を参考に、主な要素を以下に示します。

要素	説明	具体例（週末の京都旅行プランのアイデアを求める場合）
指示 (Instruction)	AIに何をしてほしいかを明確に伝える、動詞や命令文を中心とした部分です。「〇〇を作成して」「〇〇を教えて」など、具体的な行動を促します。	「提案してください」「リストアップして」「3つ教えてください」
文脈 (Context)	指示の背景となる情報、制約条件、前提知識などを提供します。これにより、AIはより状況に適した応答を生成しやすくなります。	「週末に1泊2日で京都へ行きます」「予算は一人3万円以内です」「移動手段は新幹線を考えています」「静かなお寺巡りが好きです」
役割 (Role)	AIに特定の立場やキャラクター（例：専門家、友人など）を演じさせることで、応答のトーン、スタイル、専門性を調整します。（参考資料「Prompt Engineering」より、「Role prompting」の考え方）	「あなたは経験豊富な旅行プランナーです。」「あなたは京都に詳しい地元の人です。」
入力データ (Input Data)	AIに処理させたい具体的な情報やテキストです。常に必要というわけではありませんが、特定の情報を基に応答を生成させたい場合に用います。	（例：過去の旅行アンケート結果を提示し）「このアンケート結果を基に、おすすめのスポットを教えてください。」
出力形式 (Output Format)	AIにどのような形式で答えてほしいかを具体的に指定します。これにより、情報を整理したり、後続の処理で使いやすくなりできます。（参考資料「Prompt Engineering」より、「Be specific about the output」の考え方）	「箇条書きをお願いします」「各プランに簡単な説明（100字程度）を付けてください」「費用、所要時間、おすすめポイントを表形式でまとめてください」

表1：プロンプトの構成要素と具体例

これらの要素を適切に組み合わせることで、AIに対してより具体的に分かりやすい指示を与えることができます。

指示の明確性がもたらす効果：なぜ「具体的に」が大切なのか？

AI、特にLLMとおしゃべりする時、一番大切なことのひとつは「具体的に、そして明確に伝えること」です。なぜなら、AIは人間のように相手の表情を読んだり、言外のニュアンスを「察する」ことが苦手だからです。参

考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」の「Best Practices」セクションにある「Be specific about the output（出力について具体的に）」や「Design with simplicity（シンプルに設計する）」という考え方は、まさにこの点を強調しています。曖昧な指示は、AIを混乱させ、期待しない結果や意図しない応答を生んでしまう可能性があります。

例えば、今日の天気を知りたい場合を考えてみましょう。

比較例1：曖昧なプロンプト vs 明確なプロンプト（今日の天気）

曖昧なプロンプト例：「天気教えて」

AIの応答イメージ：これだけだと、AIは「どこの天気ですか？」「いつの天気予報をご希望ですか？」と聞き返してきたり、あるいは現在地の一般的な天気を大まかに答えたりするかもしれません。情報が不足しているため、AIは的確な応答を生成するのが難しいのです。

明確なプロンプト例：「今日の東京の天気予報を、最高気温、最低気温、降水確率を含めて、箇条書きで教えてください。」

AIの応答イメージ：「承知しました。今日の東京の天気予報です。」

- 最高気温: 25℃
- 最低気温: 18℃
- 降水確率: 20%

といった形で、ユーザーが求めている情報を正確に、かつ指定された形式で提供してくれる可能性が高まります。

この比較から分かるように、明確なプロンプトは、**場所（東京）、時間（今日）、欲しい情報の種類（最高/最低気温、降水確率）、そして答えの形（箇条書き）**を具体的に指定しています。これにより、AIは迷うことなく、ユーザーが本当に知りたい情報を提供できるのです。曖昧なプロンプトでは、AIが何をすべきか判断できず、何度もやり取りが必要になったり、期待外れの応答になったりする可能性が高まります。

このように、指示の明確性は、AIとのコミュニケーションをスムーズにし、より質の高い結果を得るための基本中の基本と言えるでしょう。

プロンプト作成の基本テクニック：AIに「伝わる」指示のコツ

AI、特に大規模言語モデル（LLM）から期待通りの応答を引き出すためには、プロンプトの書き方にいくつかのコツがあります。この章では、AIに「伝わる」指示を出すための基本的なテクニックを紹介します。これらのテクニックは、特別な専門知識がなくてもすぐに実践できるものばかりです。

具体的な指示を出すコツ：「察して」ではなく「伝えて」

AIは人間のように「空気を読む」ことや「行間を読む」ことが苦手です。そのため、AIに対する指示は、できる限り具体的であることが重要です。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」のベストプラクティスにある「Be specific about the output（出力について具体的に）」や「Use Instructions over Constraints（制約よりも指示を使う）」といった考え方が参考になります。

- **5W1Hを意識する：**プロンプトに「いつ（When）」「どこで（Where）」「誰が（Who）」「何を（What）」「なぜ（Why）」「どのように（How）」といった情報を加えることで、指示が具体的になります。例えば、「記事を書いて」と頼むよりも、「〇〇というテーマについて、小学生にも分かるように、メ

リットとデメリットをそれぞれ3点ずつ挙げ、800字程度で解説記事を作成してください」と指示する方が、AIは格段に理解しやすくなります。

- **曖昧な言葉を避ける**：「いい感じに」「よしなに」「適当に」といった曖昧な表現は避け、具体的な動詞や名詞を使いましょう。AIはこれらの言葉を人間と同じように解釈できません。
- **肯定的な指示を心がける**：「〇〇しないでください」という否定的な制約よりも、「〇〇してください」という肯定的な指示の方が、AIが何をすべきかを理解しやすい傾向があります。例えば、「ゲームの話題は避けてください」と言う代わりに、「ビジネス関連の話題に限定して、〇〇について3つのアイデアを出してください」と指示する方が効果的な場合があります。

要するに、「AIに察してもらおう」と期待するのではなく、「AIに明確に伝える」ことを常に意識するのがコツです。

効果的なコンテキスト（背景情報）の与え方：AIにも「状況説明」が必要

AIがより適切な応答を生成するためには、指示内容だけでなく、その指示が出された「背景」や「文脈（コンテキスト）」を伝えることも非常に有効です。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」における「Contextual prompting」の考え方は、この重要性を示しています。

コンテキストとは、AIが応答を生成する上で役立つあらゆる関連情報のことです。例えば、特定の状況、目的、制約条件、聞き手の知識レベルなどがコンテキストになり得ます。これらをプロンプトに含めることで、AIはユーザーの意図をより深く理解し、よりパーソナルで、状況に適した応答を生成しやすくなります。

具体例：ビジネスメールの作成依頼

コンテキストが不十分なプロンプト：「お断りのメールを書いてください。」

これでは、AIは何を、誰に、どのようなニュアンスでお断りすれば良いのか全く分かりません。

コンテキストが豊富なプロンプト：「取引先の〇〇株式会社 山田部長から、来週開催される△△セミナーへの参加依頼のメールをいただきました。大変申し訳ないのですが、その日は既に重要な社内会議が入っており、どうしても都合がつかいません。今後の良好な関係を維持したいので、失礼のないよう丁寧にお断りする返信メールの文案を作成してください。セミナーの成功を祈る一言も添えたいです。」

このプロンプトでは、相手（山田部長）、目的（セミナー参加のお断り）、理由（先約）、伝えたいニュアンス（丁寧さ、良好な関係維持）、追加したい内容（セミナー成功を祈る言葉）といった具体的なコンテキストが提供されています。これにより、AIは状況に即した適切なメール文面を生成しやすくなります。

役割（ロール）を指定する：AIを「〇〇の専門家」に任命しよう

AIに特定の「役割（ロール）」を与えることは、応答のスタイル、トーン、専門性、視点などを効果的にコントロールするテクニックです。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」の「Role prompting」セクション (Table 5, 6など) で紹介されているように、AIに「あなたは〇〇です」と役割を定義することで、その役割になりきった応答を引き出すことができます。

例えば、「あなたは経験豊富な編集者です」「あなたは小学校の先生です」「あなたはフレンドリーな旅行アドバイザーです」といった役割を設定できます。これにより、AIはより一貫性のある、特定のペルソナに基づいた応答を生成するようになります。

例2：役割設定プロンプトのテンプレートと具体例

役割設定プロンプトの基本テンプレート：

あなたは [役割名] です。[役割に関する詳細や特徴、話し方など]。
以下の [タスクや質問] について、[役割名] として [具体的な指示内容] を教えてください。

具体例：エネルギー問題について子供向けに説明してもらう

あなたは経験豊富で、子供たちに物事を教えるのがとても上手な小学校の先生です。難しい言葉は使わず、いつも笑顔で、例え話をたくさん使って説明します。
最近よく聞く「再生可能エネルギー」について、小学3年生にも分かるように、私たちの生活とどう関係があるのか、優しく説明してください。

期待される効果： このように役割を設定することで、AIは専門用語を避け、比喻を用いたり、語りかけるような口調で説明したりするなど、「小学校の先生」らしい応答をする可能性が高まります。例えば、「太陽さんや風さんの力を借りて電気を作ることなんだよ」といった表現が期待できるかもしれません。

望ましい出力形式を指定する：欲しい答えの「形」を伝えよう

AIからの応答を、特定の「形」や「フォーマット」で受け取りたい場合があります。例えば、情報をリスト形式で見たい、箇条書きで整理してほしい、あるいは表形式で比較したい、といった場合です。このような要望は、プロンプト内で明確に指示することで実現できます。参考資料「Prompt Engineering (22365_3 Prompt Engineering_v7(1))」の「Best Practices」にある「Experiment with output formats (出力形式を試す)」や「Be specific about the output (出力について具体的に)」¹⁾、Table 4のJSON形式での出力例などがこの考え方を示しています。

出力形式を指定することで、AIの応答を後から人間が見やすくしたり、他のプログラムで処理しやすくしたりするメリットがあります。

図2：出力形式指定プロンプトの例（表形式での比較）

プロンプト例：

リンゴとミカンについて、それぞれの主な栄養素、旬の時期、代表的な国内品種を3つずつ挙げ、それらをまとめた表を

期待されるAIの出力（イメージ）：

果物	主な栄養素	旬の時期	代表的な国内品種
リンゴ	ビタミンC、カリウム、食物繊維、ポリフェノール	秋～冬（品種による）	ふじ、つがる、王林
ミカン	ビタミンC、β-クリプトキサンチン、食物繊維	冬	温州みかん、伊予柑、ポンカン

説明：上記のようにプロンプトで「表を作成してください」と具体的に指示することで、AIは情報を整理し、指定された形式（この場合は表形式）で応答します。これにより、リンゴとミカンの特徴を一覧で比較しやすくなります。

他にも、「箇条書きで3つのポイントを教えてください」「メリットとデメリットをそれぞれ説明し、最後に結論を述べてください」といったように、文章の構成を指定することも可能です。JSONやXMLといった特定のデータ形式での出力を求めることも、より高度な活用では有効です。

プロンプトエンジニアリングの応用テクニック入門：一歩進んだ使い方

基本的なプロンプト作成のコツを掴んだら、次はいよいよAIの能力をさらに引き出すための応用テクニックに挑戦してみましょう。ここでは、特にAI初心者や非エンジニアの方にも比較的取り組みやすい、いくつかの代表的な応用テクニックを紹介します。

Few-shot プロンプティング：AIに「お手本」を見せる

「Few-shot（フューショット）プロンプティング」とは、**AIに対して、いくつかの具体的な質問（入力）とそれに対する望ましい回答（出力）のペア（例＝ショット）をプロンプト内で「お手本」として提示する手法**です。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」の「One-shot & few-shot」セクション (Table 2など) で詳しく説明されています。

お手本を1つだけ提示する場合は「One-shotプロンプティング」、複数提示する場合は「Few-shotプロンプティング」と呼ばれます。AIはこれらのお手本から、あなたが期待する回答のパターン、スタイル、あるいは思考の道筋などを学習し、新しい質問に対しても同様の質の高い応答を生成しやすくなります。

このテクニックは特に、AIに特定のフォーマットで回答してほしい場合や、独自のニュアンスを伝えたい場合、あるいは少し複雑な指示を理解させたい場合に有効です。例えば、顧客からの問い合わせメールの内容を「ポジティブ」「ネガティブ」「ニュートラル」に分類する作業をAIにさせたい場合、いくつかのお手本を見せることで、AIの分類精度を向上させることができます。

例3：Few-shotプロンプトの構造（感情分析の例）

以下は、文章の感情を分析させるためのFew-shotプロンプトの構造例です。AIに「このように考えて、このように答えてほしい」というお手本を示します。

以下は、顧客からのフィードバック文とその感情分類の例です。
このパターンに従って、最後のフィードバック文の感情を分類してください。

フィードバック文：「この新しいアプリ、本当に使いやすくて感動しました！素晴らしいです。」
感情：ポジティブ

フィードバック文：「先日購入した製品がすぐに壊れてしまい、非常にがっかりしています。」
感情：ネガティブ

フィードバック文：「特に可もなく不可もなくだと思います。」
感情：ニュートラル

フィードバック文：「明日の新機能リリース、本当に心待ちにしています！きっと素晴らしいものになるでしょう！」
感情：

このプロンプトに対して、AIは最後のフィードバック文の感情を「ポジティブ」と回答することが期待されます。お手本を通して、AIはどのような基準で感情を分類すればよいかを学習するのです。

Chain-of-Thought (CoT) プロンプティング：「考え方」をAIに示す

「Chain-of-Thought（チェイン・オブ・ソート、CoT）プロンプティング」とは、**複雑な質問や推論が必要な問題に対して、AIに最終的な答えだけを求めるのではなく、その答えに至るまでの中間的な思考プロセスや手順を段階的に説明させる（または例示する）手法**です。日本語では「思考の連鎖」とも訳されます。参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」の「Chain of Thought (CoT)」セクション (Table 12, 13など) でこのテクニックが紹介されています。

人間が難しい問題を解くときに、いきなり答えを出すのではなく、途中の計算を書いたり、考えを整理したりするのと同じように、AIにも「考えながら」答えを出させるイメージです。プロンプトに「ステップバイステップで考えて説明してください」や「論理的な手順を示してください」といった指示を加えるだけでも、CoTの効果が得られることがあります（これは「Zero-shot CoT」と呼ばれます）。

CoTプロンプティングを用いることで、AIはより論理的で、矛盾の少ない、正確な回答を導き出しやすくなります。また、私たち人間も、AIがどのような思考プロセスを経てその結論に至ったのかを理解しやすくなり、AIの回答の信頼性を判断する助けにもなります。

例：簡単な計算問題（姉の年齢を求める）

プロンプト：

私が3歳のとき、私の姉は私の年齢のちょうど3倍でした。私が現在20歳だとすると、姉は今何歳ですか？ 答えだけでなく、どのように考えたかもステップバイステップで説明してください。

期待されるAIの応答（思考プロセスを含む）：

承知しました。ステップバイステップで考えてみましょう。

1. あなたが3歳だったとき、お姉さんの年齢はあなたの年齢の3倍だったので、 $3歳 \times 3 = 9歳$ でした。
2. その時点で、あなたとお姉さんの年齢差は $9歳 - 3歳 = 6歳$ です。お姉さんの方が6歳年上ということになります。
3. あなたが現在20歳なので、お姉さんの現在の年齢は、あなたの年齢に年齢差を加えたものになります。つまり、 $20歳 + 6歳 = 26歳$ です。

したがって、お姉さんの現在の年齢は26歳です。

このように、思考のステップを明示させることで、AIは複雑な問題に対してもより正確にアプローチできるようになります。

Zero-shot プロンプティングとの使い分け：状況に応じたテクニック選択

これまで紹介したテクニックと、最も基本的な「Zero-shotプロンプティング」をどのように使い分ければよいのでしょうか？ それぞれの特徴と適した場面を理解することで、より効果的にAIを活用できます。

- **Zero-shot プロンプティング（ゼロショット）**：AIに事前の例（ショット）を与えずに、タスクの指示のみで応答させる最もシンプルな方法です。（参考資料「Prompt Engineering」より、「General prompting / zero shot」）LLMが持つ一般的な知識や能力で対応できる比較的簡単なタスクや、自由な発想、創造的な出力を求める場合に有効です。例えば、「面白いキャッチコピーを考えて」といった場合です。
- **Few-shot プロンプティング（フューショット）**：AIに特定のタスクの例（お手本）を1つまたは少数示すことで、期待する応答のパターンやスタイルを学習させます。出力形式を整えたい場合や、タスクが少し複雑でAIに意図を正確に伝えたい場合に有効です。
- **Chain-of-Thought (CoT) プロンプティング（思考の連鎖）**：AIに思考のプロセスを明示的に生成させることで、特に複雑な推論や計算問題、論理的な判断が必要なタスクの精度を高めます。

これらのテクニックは、状況や目的に応じて使い分けことが重要です。以下の表に、それぞれの特徴と適した場面の例をまとめました。

テクニック名	特徴	適した場面の例
Zero-shot プロンプティング	お手本なしで、指示のみ。AIの汎用的な能力に依存する。最も手軽。	簡単な質問への回答、一般的な文章生成（メール、短い説明文など）、アイデアの壁打ち、クリエイティブなテキスト生成（詩、物語の冒頭など）
Few-shot プロンプティング	1～数個のお手本を提示。AIがお手本からパターンやスタイルを学習する。	特定の書式での回答を求める場合（例：報告書の一部）、特定の文体（丁寧語、フレンドリーなど）を要求する場合、簡単な分類タスク（感情分析、スパム判定など）、特定の言葉遣いをさせたい場合

テクニック名	特徴	適した場面の例
Chain-of-Thought (CoT) プロンプティング	AIに回答に至るまでの思考プロセスや論理的な手順を明示的に生成させる。	算数や数学の問題、論理パズル、複雑な指示の理解と実行、手順の説明や計画の立案を求める場合、AIの回答の根拠を知りたい場合

表2：基本・応用テクニックの比較

最初はZero-shotで試してみて、期待通りの結果が得られなければFew-shotやCoTといったテクニックを試してみる、という進め方が効果的です。

実践演習：AI初心者・非エンジニア向けケーススタディ

これまでに学んだプロンプト作成の基本テクニックや応用テクニックを、具体的なケーススタディを通して実践してみましょう。AI初心者や非エンジニアの方でも取り組みやすい、日常的な場面やビジネスシーンでの活用例を2つ紹介します。

ケーススタディ1：日々の献立のアイデア出し

毎日の献立を考えるのは大変ですよね。冷蔵庫にある食材をAIに伝えて、簡単で美味しいレシピのアイデアを出してもらいましょう。

課題設定

冷蔵庫に「鶏むね肉」「玉ねぎ」「卵」があります。これらの食材を中心に使い、調理時間が30分以内で作れる、簡単かつ美味しい夕食の献立のアイデアを3つ提案してもらおう。さらに、それぞれの献立には簡単な作り方のポイントも添えてもらうこととします。

プロンプト設計のポイント

- 役割設定：** AIに「手際の良い料理研究家」という役割を与えることで、専門的かつ実用的なアドバイスを期待します。
- 文脈・条件指定：** 使用する主要食材（鶏むね肉、玉ねぎ、卵）、調理時間の制約（30分以内）、求める献立の数（3つ）、そして「簡単で美味しい」という質に関する要望を明確に伝えます。
- 出力形式指定：** 「箇条書きで、各献立に簡単な作り方のポイントを添えてください」と、AIの回答形式を具体的に指示します。

プロンプト例

あなたは手際の良い料理研究家です。
冷蔵庫に鶏むね肉、玉ねぎ、卵があります。
これらの食材を使い、30分以内に作れる簡単で美味しい夕食の献立のアイデアを3つ提案してください。
それぞれの献立には、簡単な作り方のポイントも添えて、箇条書きをお願いします。

期待されるAIの出力（例）

AIからの応答として、以下のような献立案が期待できます。

承知いたしました！「鶏むね肉」「玉ねぎ」「卵」を使った、30分以内で作れる簡単美味しい献立ですね。こちらはいかがでしょう？

1. 親子丼

- **ポイント**：玉ねぎと鶏むね肉をだし汁で煮て、溶き卵でとじるだけの定番料理です。鶏むね肉は薄めにそぎ切りにすると火の通りが早く、柔らかく仕上がります。三つ葉や刻み海苔を最後に添えると風味も彩りもアップします。

2. 鶏むね肉と玉ねぎのふんわり卵炒め

- **ポイント**：鶏むね肉は一口大に切り、少量の酒と片栗粉を揉み込んでおくとパサつかずジューシーに。玉ねぎと一緒に炒め、最後に溶き卵を加えて手早く混ぜ合わせれば完成です。醤油、みりん、砂糖で少し甘辛い味付けがおすすめです。

3. 鶏むね肉のピカタ オニオンスライス添え

- **ポイント**：鶏むね肉を薄くそぎ切りにし、塩胡椒で下味をつけた後、小麦粉、溶き卵の順にくぐらせてフライパンで焼きます。玉ねぎは薄切りにして水にさらし、辛味を抜いてからピカタに添えると、さっぱりといただけます。ケチャップやレモンを絞っても美味しいです。

イラスト2：ケーススタディ1のプロンプトとAI応答の例（チャット画面風）

あなた（ユーザー）:

あなたは手際の良い料理研究家です。冷蔵庫に鶏むね肉、玉ねぎ、卵があります。これらの食材を使い、30分以内に作れる簡単で美味しい夕食の献立のアイデアを3つ提案してください。それぞれの献立には、簡単な作り方のポイントも添えて、箇条書きをお願いします。

AIシェフ (AI):

かしこまりました！鶏むね肉、玉ねぎ、卵を使った30分以内で作れる簡単美味しい献立ですね。こちらはいかがでしょう？

1. 親子丼

ポイント：玉ねぎと鶏肉をだしで煮て、卵でとじるだけ。七味唐辛子を振ると風味アップ。

2. 鶏むね肉と玉ねぎの卵炒め

ポイント：鶏むね肉はそぎ切りにして片栗粉をまぶすと柔らかく仕上がる。醤油とみりんできれい味付け。

3. 鶏むね肉のピカタ オニオン添え

ポイント：鶏むね肉を薄く開いて卵液にくぐらせて焼く。薄切り玉ねぎと一緒に炒めて添える。

ユーザーが具体的な食材や調理条件を指定してプロンプトを入力すると、AIがその条件に合った献立案を具体的なポイントと共に応答として返します。これにより、日々の献立作りが少し楽になるかもしれません。

ケーススタディ2：ビジネスメールの件名作成

ビジネスメールでは、相手に内容を的確に伝え、かつ失礼のない件名が重要です。AIにメールの目的を伝えて、適切な件名の案を出してもらいましょう。

課題設定

取引先である「株式会社ネクストステップ」の営業部長「山田様」に、自社の新製品「スマートプランナーPro」のサンプルを送付したことを知らせるメールを送りたい。丁寧な印象を与え、かつメールの内容が一目でわかるような件名を3案提案してもらおう。

プロンプト設計のポイント

- **役割設定**：AIに「経験豊富なビジネスアシスタント」という役割を与えます。これにより、ビジネスマナーに沿った提案が期待できます。
- **文脈・条件指定**：メールの主要な目的（新製品「スマートプランナーPro」のサンプル送付のお知らせ）、相手（取引先の株式会社ネクストステップ 山田部長）、求める件名の数（3案）、そして件名に求める要素（丁寧さ、内容の分かりやすさ）を具体的に伝えます。
- **出力形式指定**：「提案する件名は、番号付きリストでお願いします」と、回答の形式を指定します。

プロンプト例

あなたは経験豊富で、**अप्राति**（きわめて）丁寧なビジネスアシスタントです。
取引先である「株式会社ネクストステップ」の営業部長「山田様」宛に、弊社の新製品「スマートプランナーPro」のサンプルをこのメールの件名として、山田様が内容を一目で理解でき、かつ丁寧さが伝わるものを3案提案してください。
提案は番号付きリストでお願いします。それぞれの案には、なぜその件名が良いと考えるかの簡単な理由も添えてください。

期待されるAIの出力（例）

AIからの応答として、以下のような件名案とその理由が期待できます。

かしこまりました。「株式会社ネクストステップ」営業部長 山田様への新製品サンプル送付に関するメールの件名案ですね。丁寧かつ内容が分かりやすいものを3案、理由と共に提案いたします。

- 件名案：**【〇〇株式会社より】新製品「スマートプランナーPro」サンプル送付のお知らせ
 - 理由：**送信元とメールの主目的（新製品サンプルの送付）が明確で、ビジネスメールの件名として一般的かつ丁寧な形式です。製品名も具体的に記載されているため、内容の推測が容易です。
- 件名案：**新製品「スマートプランナーPro」サンプルをお送りいたしました（〇〇株式会社 山田様）
 - 理由：**より直接的にサンプルを送付したことを伝えつつ、括弧書きで社名と宛名を補足することで、受信者が自分宛の重要な連絡であると認識しやすくしています。「お送りいたしました」という表現で丁寧さも保っています。
- 件名案：**（〇〇株式会社）「スマートプランナーPro」サンプル発送のご連絡とご挨拶
 - 理由：**「ご連絡とご挨拶」という言葉を加えることで、単なる通知だけでなく、相手への配慮やコミュニケーションの意図も感じさせることができます。サンプル発送という事実と、ビジネス上の丁寧な挨拶を両立させたい場合に適しています。

これらのケーススタディを通じて、プロンプトを工夫することで、AIがより具体的に役立つ情報を提供してくれることを体験してみてください。最初は簡単なプロンプトから始め、徐々に条件を加えたり、役割を設定したりしながら、AIとの対話を楽しんでみましょう。

まとめと次のステップ

本資料では、プロンプトエンジニアリングの基本的な概念から、AI初心者や非エンジニアの方でもすぐに実践できるテクニック、そして簡単な応用例までを解説してきました。AI、特に大規模言語モデル（LLM）は、私たちの働き方や学び方、さらには日常生活における情報との関わり方を大きく変える可能性を秘めたテクノロジーです。そして、その力を最大限に引き出すための「カギ」となるのが、効果的なプロンプトを作成するスキル、すなわちプロンプトエンジニアリングです。

本資料の重要ポイント振り返り

ここで、本資料で学んだ特に重要なポイントを振り返ってみましょう。

- 💡 プロンプトはAIへの「指示書」：** AIに何をしてほしいか、何について答えてほしいかを伝えるための言葉や文章がプロンプトです。その「指示書」の書き方一つで、AIの応答の質や内容が大きく変わります。
- 💡 LLMは言葉を予測するAI：** 大規模言語モデル（LLM）は、膨大なテキストデータから言葉のパターンを学習し、次に来る単語（トークン）を予測することで文章を生成します。人間のように「理解」しているわけではなく、あくまで確率に基づいた予測を行っています。
- 💡 プロンプトの構成要素を意識する：** 効果的なプロンプトは、多くの場合「指示」「文脈」「役割」「（入力データ）」「出力形式」といった要素で構成されます。これらを意識的に組み合わせることが重要です。

- 💡 **基本テクニックの活用：**「具体的で明確な指示を出す」「適切な文脈を提供する」「AIに役割を与える」「望ましい出力形式を指定する」といった基本テクニックは、AIとのコミュニケーションを円滑にします。
- 💡 **応用テクニックでさらに深く：**「Few-shotプロンプティング」でお手本を示したり、「Chain-of-Thoughtプロンプティング」で思考のプロセスを促したりすることで、より複雑なタスクにもAIを活用できるようになります。
- 💡 **試行錯誤が上達への道：**最初から完璧なプロンプトを作るのは難しいものです。まずは簡単なプロンプトから始めて、AIの応答を見ながら少しずつ内容を修正したり、条件を加えたり、テクニックを試したりする「試行錯誤」が最も重要です。

プロンプトエンジニアリング活用のヒント

プロンプトエンジニアリングは、特別なスキルではなく、日々のちょっとした工夫で誰でも始めることができます。以下に、日常業務や個人の活動でAIを活用するためのヒントをいくつか挙げます。これらは、参考資料「Prompt Engineering ([22365_3_Prompt_Engineering_v7\(1\)](#))」の「Best Practices」セクションで述べられている「Provide examples（具体例を提供する）」「Design with simplicity（シンプルに設計する）」「Be specific about the output（出力について具体的に指示する）」などの原則にも通じるものです。

- **日常の調べ物をAIに：**インターネットで検索する代わりに、知りたいことを具体的な質問（プロンプト）としてAIに投げかけてみましょう。単なるキーワード検索よりも、対話形式でより深い情報や解説が得られることがあります。
- **文章作成のパートナーとして：**メールの下書き、ブログ記事の構成案、報告書の要点整理、プレゼンテーションのアイデア出しなど、様々な文章作成の場面でAIに手伝ってもらいましょう。「たたき台」を作ってもらっただけでも、作業効率が格段に上がります。
- **学習のサポート役として：**難しい専門用語や歴史上の出来事について、分かりやすい言葉で解説を求めてみましょう。長文の資料を要約させたり、学習内容に関するクイズを出してもらったりするのも効果的です。
- **創造的な作業の刺激に：**新しいキャッチコピーの提案、物語のあらすじ作成、歌詞のアイデア出しなど、クリエイティブな作業に行き詰まったときに、AIに多角的な視点からアイデアを出してもらうことで、新たなひらめきが得られるかもしれません。
- **タスクを分解する：**複雑な依頼をする場合は、一度に全てを指示するのではなく、タスクを小さなステップに分解し、段階的にプロンプトを与える方がうまくいくことがあります。

さらなる学習のために

プロンプトエンジニアリングは、実践すればするほど上達するスキルです。本資料で学んだことをきっかけに、ぜひ積極的にAIとの対話を楽しんでみてください。さらに知識を深めたい方のために、以下のような学習方法や情報源があります。

- **実際に使ってみる、たくさん試す：**これが最も重要です。ChatGPT（[Google Gemini](#)なども同様）のような対話型AIサービスを積極的に利用し、本資料で紹介したテクニックを試しながら、様々なプロンプトを作成してみてください。AIの応答を見ながら、「こうすればもっと良くなるかも」と工夫を重ねることが上達への近道です。
- **オンラインの情報源を活用する：**インターネット上には、プロンプトエンジニアリングに関する有益な記事、ブログ、動画チュートリアルなどが数多く存在します。「プロンプトエンジニアリング 初心者向け解説 最新」といったキーワードで検索すると、新しい情報や他の人の実践例が見つかるでしょう。
- **関連書籍を読む：**「プロンプトエンジニアリング入門」「AIとの対話術」といったキーワードで書籍を探してみるのも良いでしょう。体系的に知識を整理するのに役立ちます。
- **コミュニティに参加する：**AIやプロンプトエンジニアリングに関するオンラインフォーラムやSNSグループなどに参加し、他のユーザーの実践例を学んだり、情報交換をしたりするのも刺激になります。
- **本資料で参照したナレッジ資料の深掘り：**本資料作成の主要な知識源である英文資料「Prompt Engineering_v7 (22365_3_Prompt_Engineering_v7 (1)) (Author: Lee Boonstra, et al.)」には、本資料では触れられなかったさらに高度なテクニック（例：Self-consistency, Tree of Thoughts, ReActなど）や、より詳細なベストプラクティスが多数記載されています。英語の読解に抵抗がなければ、ぜひ挑戦してみてください。より深い理解が得られるはずです。

プロンプトエンジニアリングの世界は奥深く、そして日進月歩で進化しています。本資料が、皆さんのAI活用の一助となり、その面白さや可能性を感じていただくきっかけとなれば幸いです。

参考資料・情報源

本資料の作成にあたり、以下の資料を主要な知識源として参照いたしました。

- **資料ID:** 22365_3_Prompt Engineering_v7 (1)
- **資料名:** Prompt Engineering
- **著者:** Lee Boonstra, et al. (Content contributors: Michael Sherman, Yuan Cao, Erick Armbrust, Anant Nawalgaria, Antonio Gulli, Simone Cammel; Curators and Editors: Antonio Gulli, Anant Nawalgaria, Grace Mollison; Technical Writer: Joey Haymaker; Designer: Michael Lanning)
- **概要:** Googleによって提供された、プロンプトエンジニアリングの概念、様々なテクニック、ベストプラクティスについて詳述したホワイトペーパー。

本資料は、上記のホワイトペーパーに記載されている情報を基に、AI初心者および非エンジニアの読者に向けて、内容を平易に再構成し、解説を加えたものです。

追加調査データについて：

本資料作成においては、現在の日時（2025年05月26日）から遡って1ヶ月以内の最新データを積極的に利用する指示がありましたが、主に上記の提供ナレッジ資料の内容を基盤としています。外部の最新情報を取り入れる際には、ナレッジ資料の基本的な原則や技術解説との整合性を慎重に確認し、矛盾が生じないよう、また情報の信頼性を損なわないよう留意する必要があります。今回の資料では、主に提供ナレッジの普遍的な概念とテクニックの解説に焦点を当てています。

さらなる学習や最新情報の収集にあたっては、信頼できる学術論文データベース（例：[arXiv](#)）、大手AI研究機関（OpenAI, Google AI, Meta AIなど）の公式ブログや発表、専門的な技術メディアの記事などを参照することをお勧めします。