電通総研

応用機械学習と人工知能セミナー: AIエージェント

ヘルプデスクの事例から学ぶAIエージェント

2024年07月18日(木)

電通総研 太田真人

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

自己紹介

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

太田真人（Masato Ota）

株式会社電通総研 Xイノベーション本部 AIトランスフォーメンションセンター

- SIerの会社でAI製品開発、技術検証やPoC案件に従事

X: @ottamm_190

人間とシステムとAI のインタラクション技術が好き

- LLMに基づく自律型エージェント
- 予測の不確実性
- Human in the Loop & XAI

技術発信（masatoto）

- Weekly AI Agents News!
- LLMマルチエージェントを俯瞰する
- ICLR2024 LLMエージェントの研究動向

本日お話しすること

- **AIエージェントとは何か**
- **AIエージェントをなぜ開発するのか**
- **AIエージェントの開発方法とは**
- **AIエージェントの課題とは**

開発者目線でAIエージェントを広く見渡します。

ヘルプデスクでの応用事例も紹介しながら理解の解像度を高めます。

研究動向に関しては公開している資料をご覧ください。

電通総研

AIエージェントとは何でしょうか

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

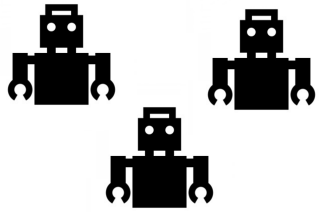
最初にAIエージェントの応用を見渡してみる

AIエージェントの研究応用例

Agentic AI System
エージェントが主体的に業務やタスクを自動化する

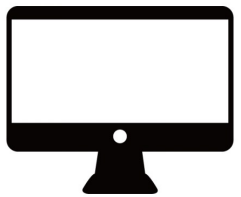
業務の自動化

Multi-Agent System
複数のエージェントが協力/競争し、シミュレーションや問題解決をする



Embodied Agents
目標に向けて環境と相互作用する身体のあるエージェント

Computer Control Agents
コンピュータ上のタスクを自動化するエージェント



AIエージェントのビジネス応用例

消費者向け
- 旅行や移動計画
- 価格比較や商品推薦
- アカウント/サブスク管理

社内/バックオフィス業務向け
- 会議の予約調整
- 法務や人事など書類作成/レビュー
- コスト管理
- 社内システムの質問応答

コア業務向け
- カスタマサポート
- ソフトウェア開発
- ビジネスデータ分析
- 特許, 文献, 企業や市場調査

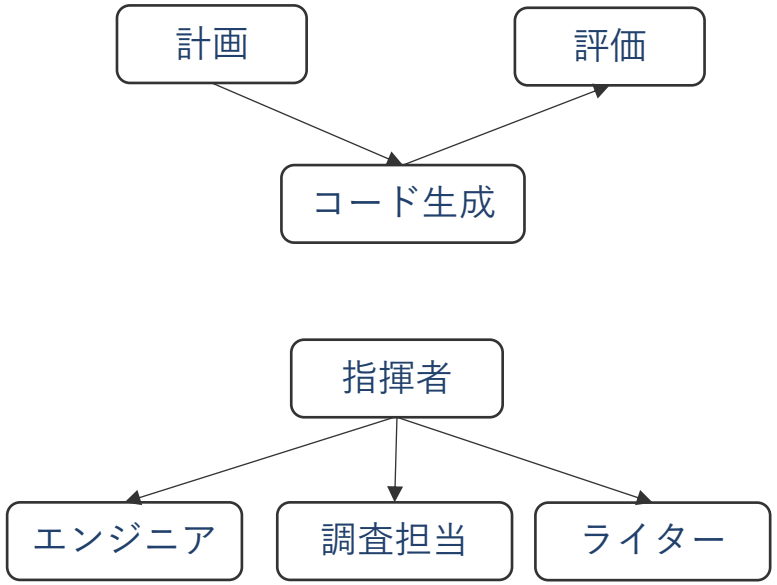
問題解決に絞ると3種類のAIエージェントの技術がある

▌様々なAIエージェントの応用事例は以下のいずれかの技術に基づく

本日取り上げる内容

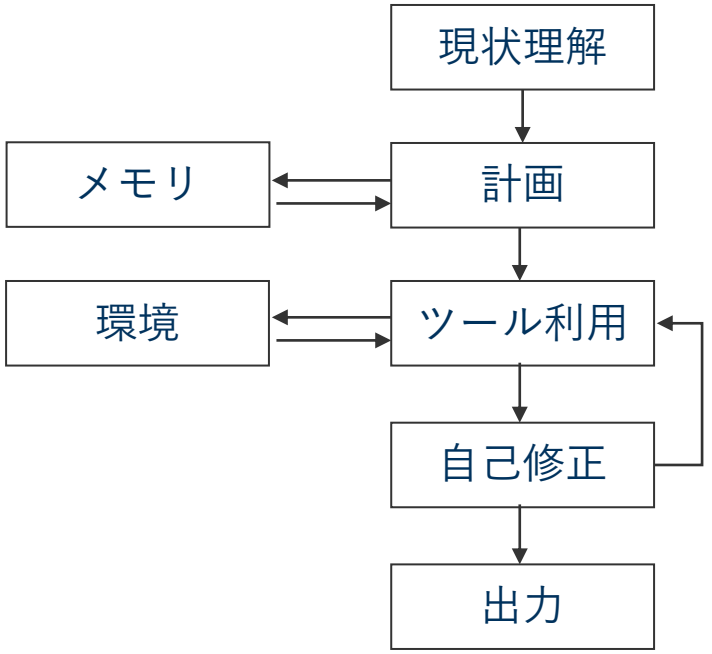
マルチエージェントの協調モデル
（Multi-Agent Collaboration）

複雑な問題を役割で分解し単純化



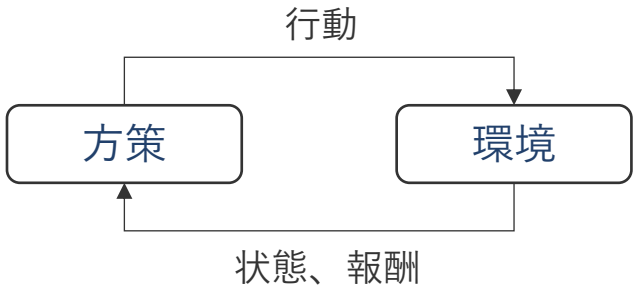
LLMに基づく自律型エージェント
（LLM-based Autonomous Agents）

エージェントアーキテクチャを構築



強化学習エージェント
（RL Agents）

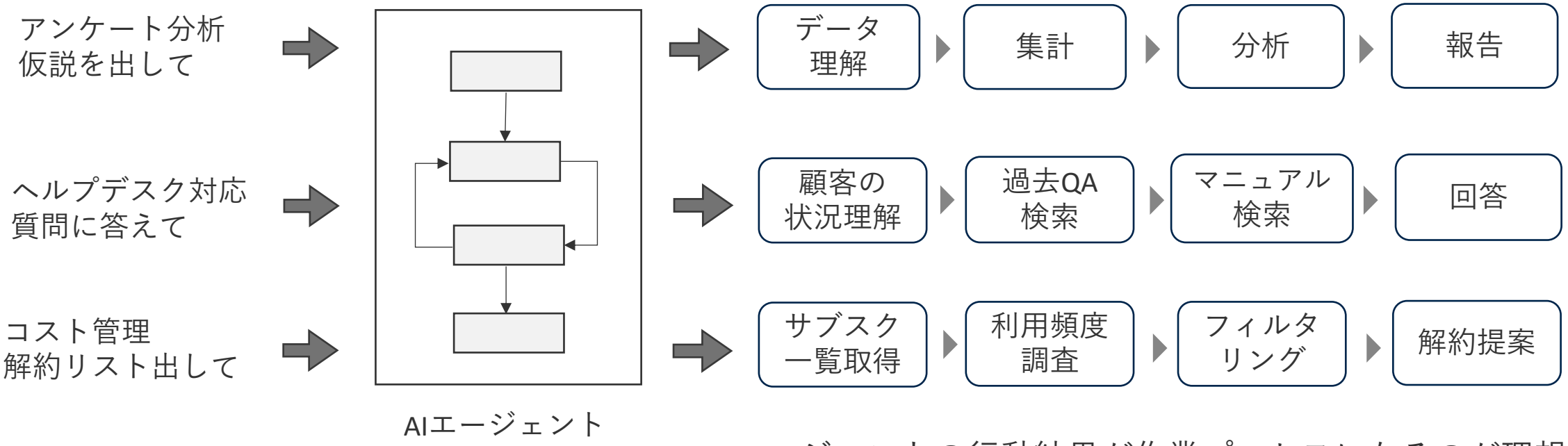
環境との相互作用から方策を学習



※ピンクがエージェントの根幹に相当する。
※エージェントは第二次AIブームでも分散人工知能（Distributed AI）として研究されていました。

AIエージェントは何ができるのが理想か

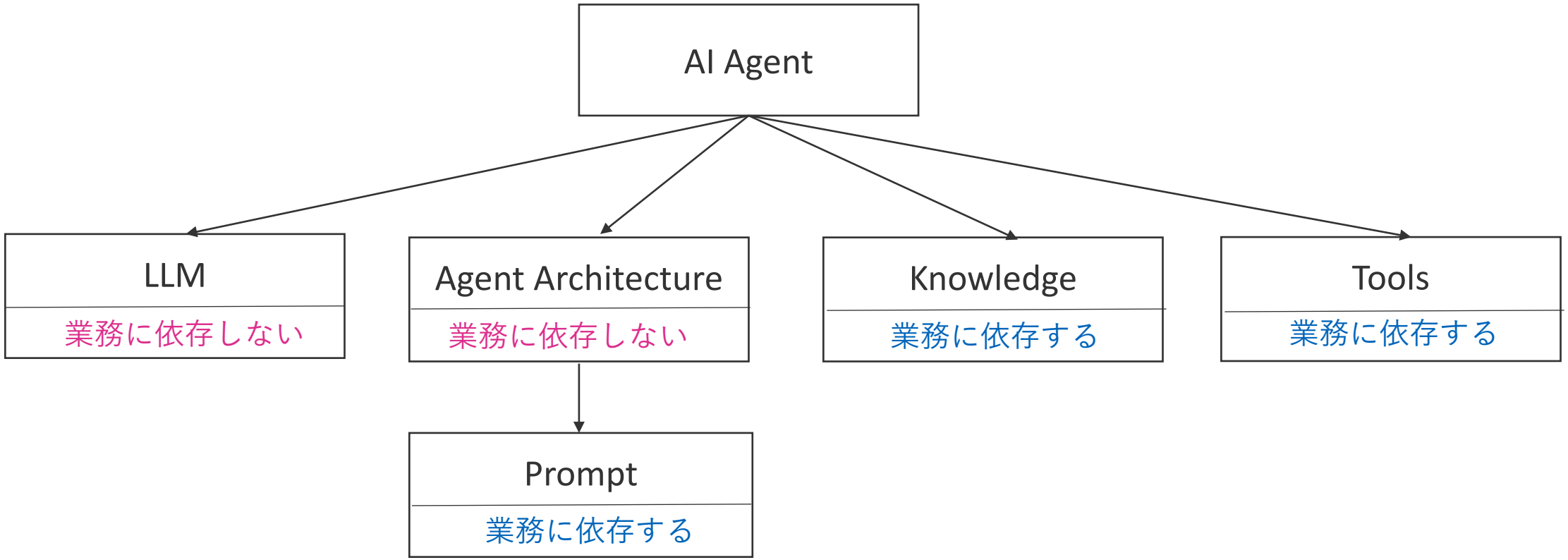
- 人間の様々なタスクの作業プロセスを自律的に遂行できる
- 業務の汎用性とタスク遂行率はトレードオフ



エージェントの行動結果が作業プロセスになるのが理想

もう少しAIエージェントの実態を開発者目線で考えてみる

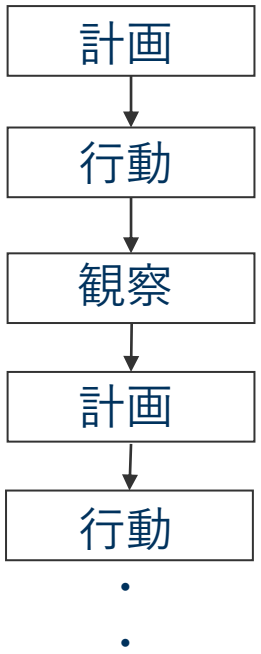
- AIエージェントは業務に依存する部分と依存しない部分に分けられる
- 業務に依存する部分をチューニングすることで業務特化にできる
- LLM やAgent Architecture は業務から切り離して汎用的に使えるようにする



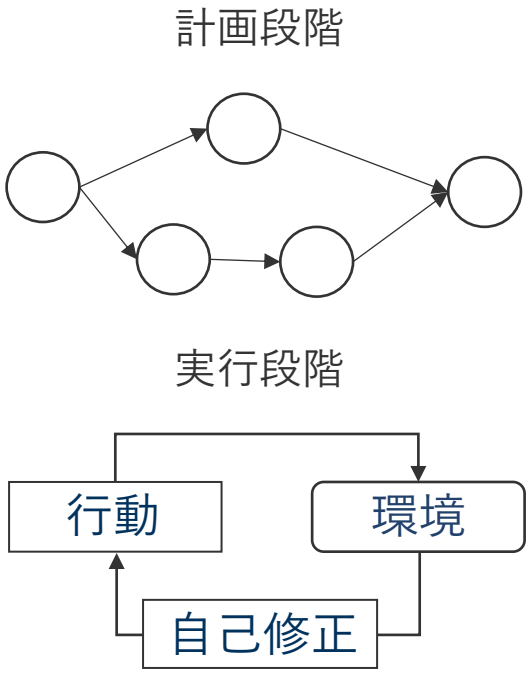
Agent Architecture とは何なのか

- Agent Architecture とは、LLMを用いた自律的に問題解決する汎用的なワークフロー
- ワークフローの構成には計画、Knowledgeを含むメモリ、Toolsを使う行動が含まれる

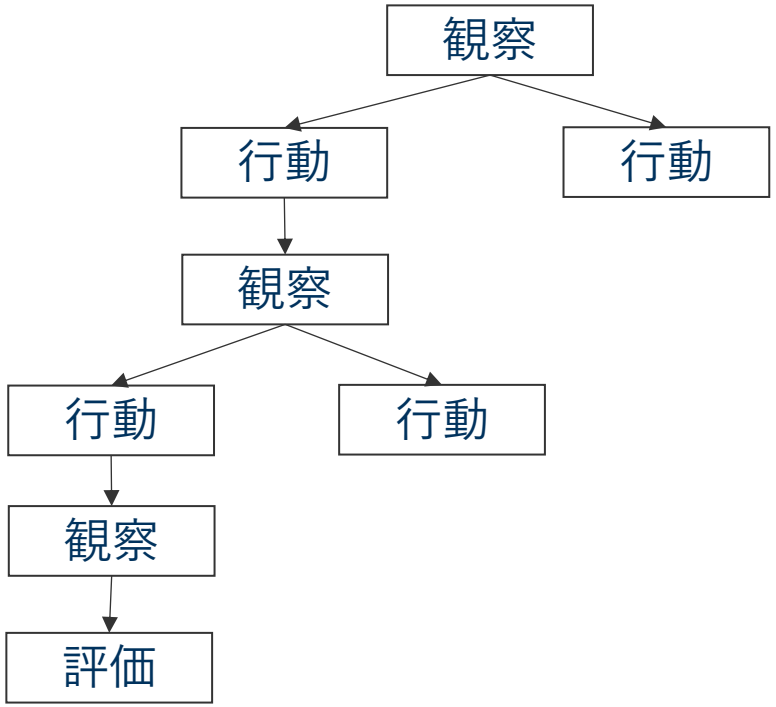
Sequential 型



Plan & Action 型



Tree 型



Sequential： ReAct (2022), Reflexion (2023)
Plan & Action：ReWoo, LLM Compiler (2023)
Tree：LATS, ToT (2023)

自律的に問題解決するとは何を意味するのか

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

- タスクの意図を理解できる
- 問題解決までの道筋を計画できる
- 自ら行動内容を決めて実行できる
- 環境から得られる情報に適応できる

上記の能力を強化するために以下の要素もかかせない

- 心の理論：相手の状況を理解する
- メモリの活用：過去の経験、ナレッジを活かして意思決定する
- 自己修正：行動や計画の誤りを自ら正す
- 自己進化：経験をもとに継続的に性能を向上させる

AIエージェントとは何か

- どの技術を前提にするかでAIエージェントの言葉の定義は変わる
- 今回のスコープでは人間の様々なタスクの作業プロセスを自律的に遂行できるソフトウェアとする
- 業務に依存しないLLMやエージェントアーキテクチャと業務に依存するツール、ナレッジで構成される
- タスクの意図を理解し、問題解決までの道筋を計画できる
- 自ら行動内容を決めて実行でき、環境から得られる情報に適応できる

なぜAIエージェントを開発するのか

AIエージェントの開発方法とは

AIエージェントの課題とは

電通総研

なぜAIエージェントを開発するのか

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

LLM, RAG と RPAワークフローはAIエージェントではない

LLM単体：プロンプトを与えてテキストを生成

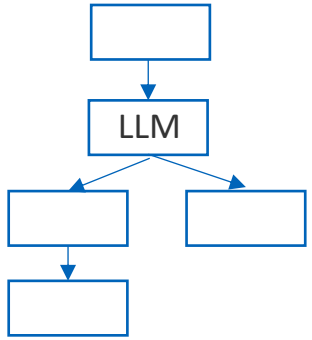
- 行動の実行はできない
- 環境の情報を取得できない

RAG：ドキュメント検索結果をテキストプロンプトに渡してテキストを生成

- タスクを遂行するまで検索し続けることができない
- 検索結果に応じて次の検索内容を決めるように適応できない
- タスクごとに検索からプロンプト代入までフローをカスタマイズする必要がある

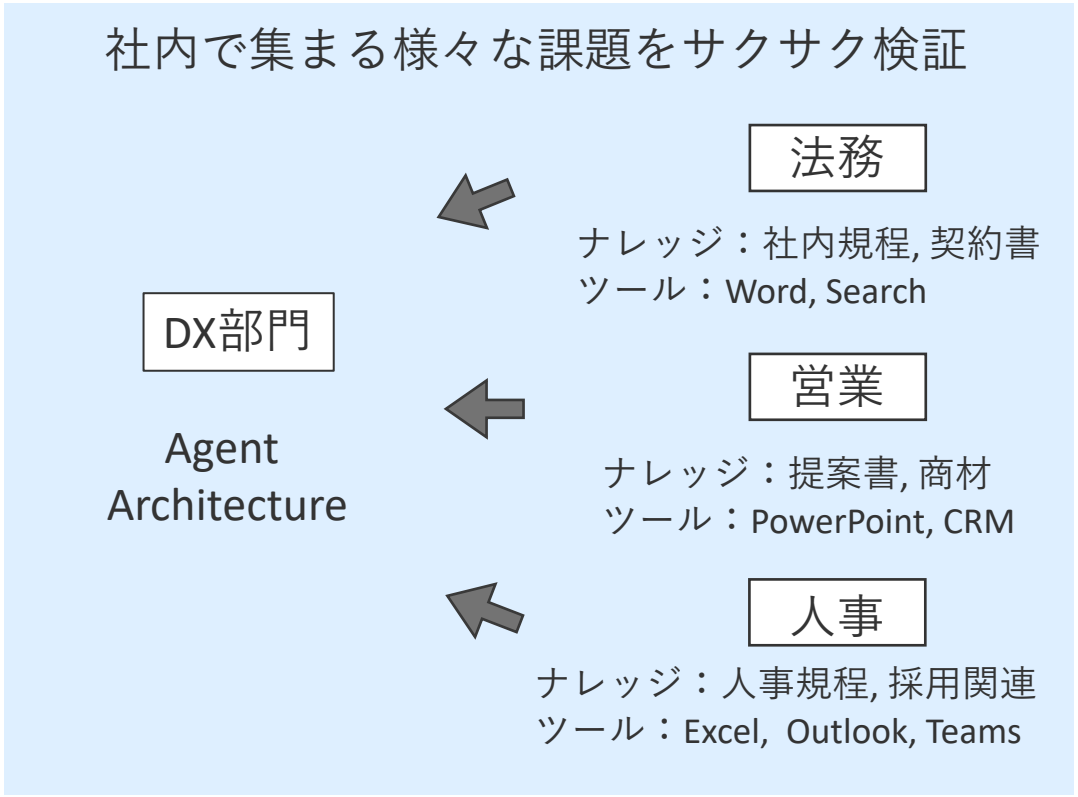
RPA Workflows: LLMを用いて業務プロセスごとにノーコードワークフローで自動化

- ワークフローを業務ごとにカスタマイズが必要
- ワークフローを作っても動作中に環境の変化に適応できない
- 複雑なタスクに対しては分岐の多い複雑なワークフローになる

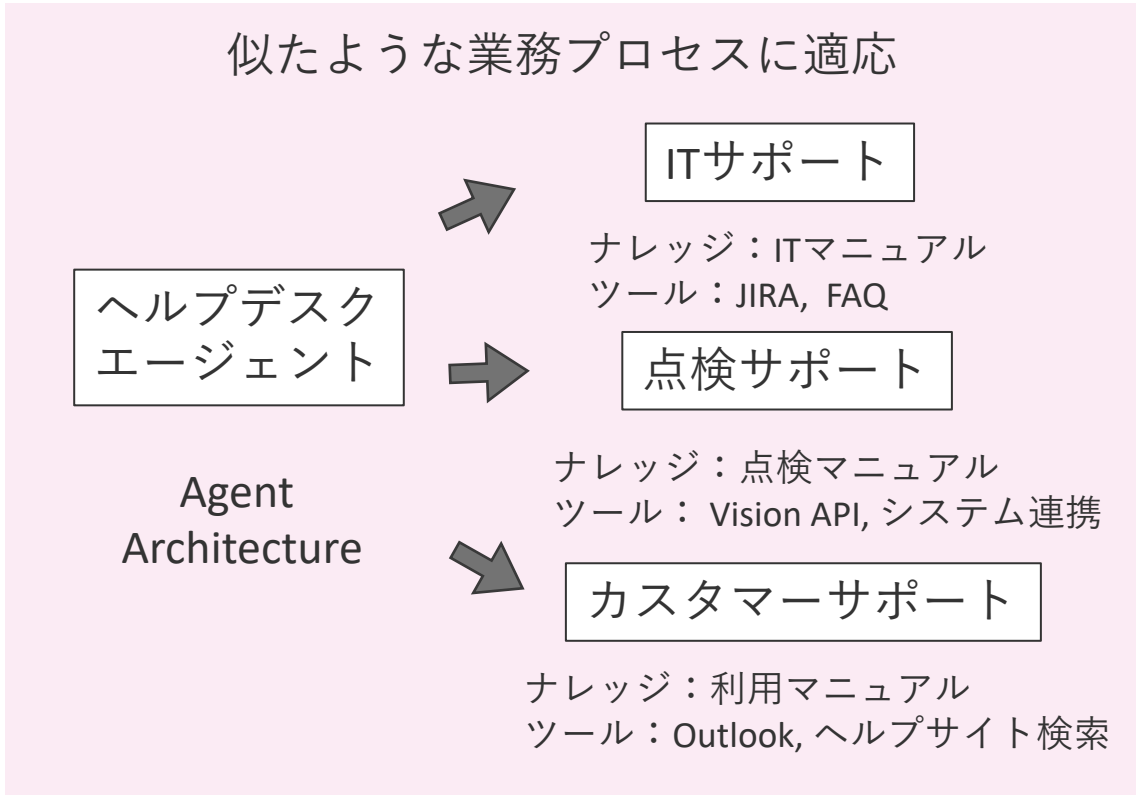


AIエージェントはカスタマイズ性と汎用性から様々な業務課題を解決できる

- エージェントアーキテクチャを一度作れば、様々な業務で汎用的に使え、PoCが高速にできる
- 最近のAIエージェントは RAG案件、ビジネスデータ分析、RPAの手法の一つになっている



個別開発の省力化



水平方向に展開

AIエージェントの開発フレームワークの充実

AITC AI TRANSFORMATION CENTER トランスフォーメーションセンター

- 市民開発：業務に依存する部分を開発する
- プロ開発：エージェントアーキテクチャから開発する

エージェントの種類	開発フレームワーク（API含む）	特徴
業務に依存する部分を与えて簡易に試せるエージェント	GPTs, Copilot Studio, Agents for Amazon Bedrock, Vertex AI Agents, Dify Agents	Agent Architecture は変更できない Promptは自動生成されることもある
LLMとAgent Architecture を選べてプロンプトも書けるエージェント	LangChain Agents, LlamaIndex Agents	Agent Architecture が色々と用意されている
Agent Architecture から作れるエージェント	LangGraph, Assistants API, LLMの生成用 API	Agent 開発に必要なスレッドや状態管理などがサポートされている
一から作れるマルチエージェント	LangGraph, AutoGen, Crew AI	エージェント間の通信設計ができる

AI Agent = LLM + Agent Architecture + Prompt + Knowledge + Tools

業務に依存しない　　業務に依存する

ここまで

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

AIエージェントとは何か

- どの技術を前提にするかでAIエージェントの言葉の定義は変わる
- 今回のスコープでは人間の様々なタスクの作業プロセスを自律的に遂行できるソフトウェアとする
- 業務に依存しないLLMやエージェントアーキテクチャと業務に依存するツール、ナレッジで構成される
- タスクの意図を理解し、問題解決までの道筋を計画できる
- 自ら行動内容を決めて実行でき、環境から得られる情報に適応できる

なぜAIエージェントを開発するのか

- 開発フレームワークの充実で市民もプロも開発の敷居が下がっている
- 業務に依存する部分の変更だけでRAG、RPAやデータ分析を高速に検証できる

AIエージェントの開発方法とは

AIエージェントの課題とは

電通総研

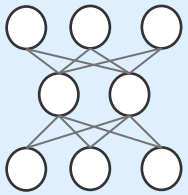
AIエージェントをどのように開発するのか

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

AIエージェントの開発方法

本日取り上げる内容

学習

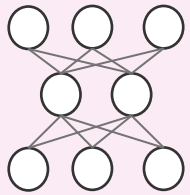


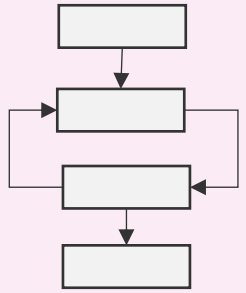
エージェント能力

LLM

LLMにエージェント能力を与える
- モデルの機構の変更（Language Action Model）
- 推論能力強化の事後学習
- function calling 機能を指示チューニング
- エージェントの振る舞いを微調整
- 計画能力にRLのアルゴリズムを適応

推論





LLM

Workflow

LLMのエージェント能力を活用する
- プロンプトエンジニアリング
- エージェントワークフロー開発
- function calling 機能のためのツール開発
- メモリ、検索システムの設計
- ナレッジの管理

AIエージェントの開発手順

1. 業務プロセスを書き出し、エージェントの理想的な行動パターンを考える
2. 行動で使うツールを作る
3. エージェントが必要となるナレッジを定義する
4. エージェントのアーキテクチャを決める
5. プロンプトエンジニアリング

AI Agent = LLM + Agent Architecture + Prompt + Knowledge + Tools

弊社で取り組んだヘルプデスクエージェントの問題設定

- ヘルプデスクの題材：弊社開発の社内利用ChatGPTやRAGソリューションの問い合わせ窓口
- エージェントのタスク：問い合わせの一次対応の回答案作成

1. 業務プロセスを書き出し、エージェントの理想的な行動パターンを考える

AITC AI TRANSFORMATION CENTER トランスフォーメーションセンター

▌担当者に「いつ、どのツールを使い、何をするのか」を聞き出す

▌エージェントの理想的な計画と行動を書き出す

理想的な計画
- 類似質問を集め、回答に必要な情報の収集

理想的な行動
- 製品の基本的な仕様はヘルプサイトを検索
- エラーメッセージは開発ドキュメントの検索
- Azureの仕組みは MS LearnをWeb検索

ヒアリング結果

作業プロセス	ツールやシステム
問い合わせ内容と添付ファイルの確認	FreshDesk
過去の問い合わせに類似するものがあるか確認	FreshDesk
ヘルプサイトに回答の記述があるか確認	ヘルプサイト
回答に必要な情報収集 • 有識者に質問 • バグなら再現 • コードを確認 • AzureのWebドキュメントを調査	teams ソースコード 開発ドキュメント MS Learnドキュメント
回答案を作成	teams
メール送信	FreshDesk
ヘルプサイトに記述	ヘルプサイト

2. 行動で使うツールを作る

AITC AI TRANSFORMATION CENTER トランスフォーメーションセンター

- ツールは、環境情報の取得、環境の状態更新、計算するもので分けられる
- Tool Calling 用の関数表記は精度に直結し、改善中は何度も書き換えることになる

Bing APIを使うPython関数
Input : query
Process : 1件検索→後処理
Output : title, html, url

Azure AI Searchを使うPython関数
Input : query (Japanese)
Process : 3文章取得
Output : (title, text)*3

Tool Calling 用の関数表記

```
{
   "type": "function",
    "function": {
         "name": ”function name",
         "description": ”function description",
         "parameters": {
               "type": "object",
               "properties": {
                     "query" : {
                           "type": "string",
                           "description": "parameter description" },
               },
         "required": ["query"] }
   }
}
```

3. エージェントが必要となるナレッジを定義する

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

▌ナレッジにはタスクを解く上で必要になる事前知識をかく

▌ナレッジの文量が増えると、検索とナレッジの更新が必要になり、管理が課題

- エキスパートシステムの時代でも知識の獲得と管理が課題だった

社内利用ChatGPTの前提知識

何のための製品か

誰が使うのか

主要な機能は何か

どのように開発されているか

プロンプト

あなたはヘルプデスクエージェントです。
ユーザーの質問に対して一次回答をします。

以下の製品に関してユーザーは質問します。

{Knowledge}

…

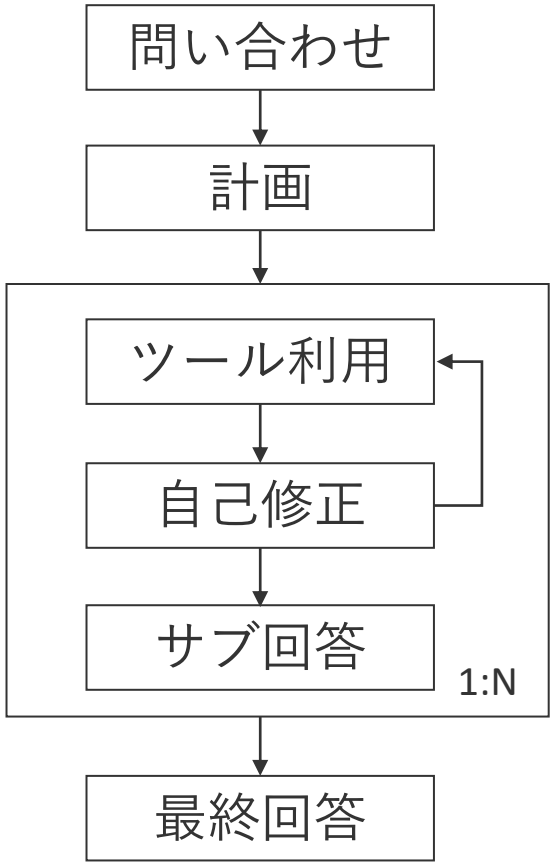
4. エージェントアーキテクチャを決める

ヘルプデスクでは Plan & Action 型を採用した

- Sequential型のReActは人間の介入できるポイントが少なく、難しいタスクで無駄に長く推論が続き、誤りが累積し、デバッグがしにくい
- Plan & Action型はPlanとAction段階それぞれで人間による動作チェックができ、難しいタスクを簡易サブタスクに分解でき、サブタスクごとに評価できる

アーキテクチャの考慮ポイント

- 計画を**静的**にするか、行動の結果で変更する動的にするか
- 計画の粒度（サブタスク間に依存関係を持たせるか、**完全に独立**にするか）
- 自己修正をどこでするか（**行動後**、サブ回答後、最終回答後）



提案エージェントのワークフロー

Tips. エージェントのアーキテクチャで参考にした論文

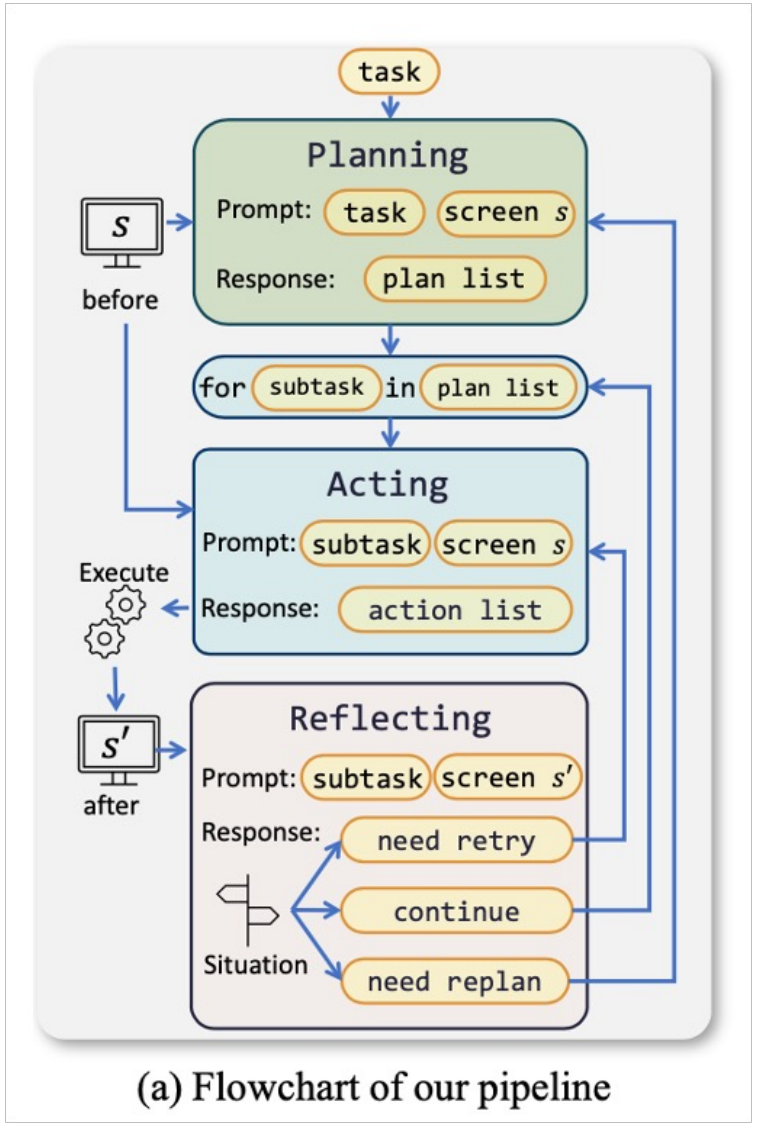
ScreenAgent : A Vision Language Model-driven Computer Control Agent

コンピュータ制御エージェント

- タスクはパワポ編集、ECサイトでの商品購入など
- ツールはマウス操作やキーボード入力
- 観測はスクショ画像
- エージェントアーキテクチャは業務に依存しないため再利用可能

エージェントアーキテクチャの特徴

- Plan & Action 型のアーキテクチャ
- Reflectionではサブタスク内容とスクショをもとに再行動、計画の続行、再計画の３種類のいずれかを生成



(a) Flowchart of our pipeline

Niu, Runliang, et al. "Screenagent: A vision language model-driven computer control agent." arXiv preprint arXiv:2402.07945 (2024).

精度を高める工夫を紹介します

Planning Prompt

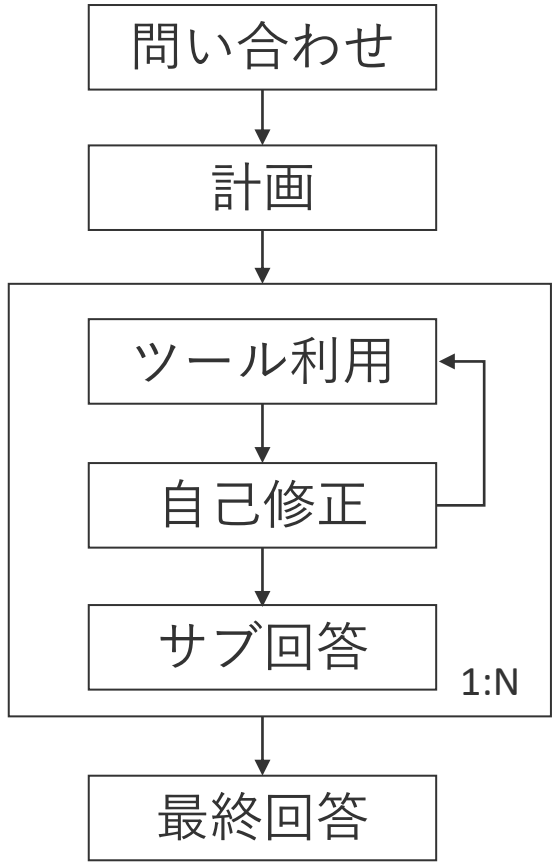
- jsonモードを使う
- 計画のサブタスク数を少なくさせる
- Reflection しやすいように、どのツールを使って何を得たいか生成させる

Tool Use Prompt

- ツールの関数名、各説明文の曖昧さをなくす
- アプリケーションごとにツールにしない
- 複数の検索インデックスは用途ごとにそれぞれ関数にする

Reflection Prompt

- ツールの実行結果からサブタスクを達成できるか評価する
- ツールが良くない場合、次のツールやパラメータ案を生成する



全体のエージェント実行フロー

Question

利用データはどこに保存されるの？
オプトアウト申請してるから安心？

Planning

subtask 1:
search_know_narrator_help_site_docsを使って'Know Narrator Chat データ保存先'を知る
subtask 2:
search_know_narrator_developer_docsを使って'Azure OpenAI Service データ取り扱い'を知る
subtask3:
search_microsoft_learn_docsを使って'Azure OpenAI Service オプトアウト申請後のデータ取り扱い'を知る

Subtask 3
Subtask 2
Subtask 1

Tool Calling

"name": "search_know_narrator_developer_docs"
"arguments": { "query" : "チャット データ 保存先"}

Tool Execution Results

[{"filename": "KnowNarratorChat_サービス仕様書.pdf", "content": ~~~~~}, {"filename": "sys_message_template.md", "content": ~~~~~}, }]

Reflection

{"status" : "OK", "advice" : "なし"}

•
•

Subtask Answer

Know Narrator Chatにおけるチャットデータの保存に関する情報は、以下の通りです。 - チャット履歴やアプリケーション内で使用されるログなど...

Answer

Know Narrator Chatにおけるチャットデータの取り扱いについてご説明いたします。 Know Narrator Chatで入力されたデータやAIからの応答データは、Azureのストレージサービスに保存されます。具体的には、アプリケーションのデータベースとしてAzure XxxDBが利用されており、チャット履歴やアプリケーション内で使用されるログなどのデータは、このデータベース内に保存されます。....

ここまで

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

AIエージェントとは何か

- どの技術を前提にするかでAIエージェントの言葉の定義は変わる
- 今回のスコープでは人間の様々なタスクの作業プロセスを自律的に遂行できるソフトウェアとする
- 業務に依存しないLLMやエージェントアーキテクチャと業務に依存するツール、ナレッジで構成される
- タスクの意図を理解し、問題解決までの道筋を計画できる
- 自ら行動内容を決めて実行でき、環境から得られる情報に適応できる

なぜAIエージェントを開発するのか

- 開発フレームワークの充実で市民もプロも開発の敷居が下がっている
- 業務に依存する部分の変更だけでRAG、RPAやデータ分析を高速に検証できる

AIエージェントの開発方法とは

1. 業務プロセスを書き出し、エージェントの理想的な行動パターンを考える
2. 業務に依存する部分のツール、ナレッジを開発する
3. エージェントのアーキテクチャを開発し、最後はプロンプトエンジニアリング

AIエージェントの課題とは

電通総研

現在のAIエージェントの技術的課題

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター

AIエージェントの技術的な課題

AIエージェントには、理解力、計画力、行動力、適応力が求められる

理解力

- ユーザーからの質問の意図の理解、長文による指示の理解
- 問題解決には必要のない背景も考慮する（フレーム問題）
- 相手の置かれている状況やユーザーの理解力を推察できていない（心の理論）

計画力

- タスクを解決するために実行可能な手順に落とし込むこと
- 過去の経験がない場合、人間のように実行手順に落とし込めない
- 同じようなサブタスクを何回も生成しがち

適応力

- 行動から得た環境の情報から柔軟に計画変更や次の行動を決めること
- 思考が進まず、似たような行動を繰り返すことが多い
- プロンプトを工夫しないと、エージェントはクエリをわずかに変えて再検索しがち

まとめ

AIエージェントとは何か

- どの技術を前提にするかでAIエージェントの定義は変わる
- 今回のスコープでは人間の様々なタスクの作業プロセスを自律的に遂行できるソフトウェアとする
- 業務に依存しないLLMやエージェントアーキテクチャと業務に依存するツール、ナレッジで構成される
- タスクの意図を理解し、問題解決までの道筋を計画できる
- 自ら行動内容を決めて実行でき、環境から得られる情報に適応できる

なぜAIエージェントを開発するのか

- 開発フレームワークの充実で市民もプロも開発の敷居が下がっている
- 業務に依存する部分の変更だけでRAG、RPAやデータ分析を高速に検証できる

AIエージェントの開発方法とは

1. 業務プロセスを書き出し、エージェントの理想的な行動パターンを考える
2. 業務に依存する部分のツール、ナレッジを開発する
3. エージェントのアーキテクチャを開発し、最後はプロンプトエンジニアリング

AIエージェントの課題とは

- 理解力、計画力、適応力に実用上に難あり。エージェント能力をLLMに与える学習にも期待！

参考文献

AIエージェントとは何か

- 研究の応用事例：電通総研, ICLR2024 LLMエージェントの研究動向
- 石田亨. (1995). エージェントを考える (< 特集>「エージェントの基礎と応用」). 人工知能, 10 (5), 663-667.
- 秋田 興一郎. (1989). エキスパート・システム: 考え方・作り方・使い方 (DSライブラリー)
- エージェントのサーベイ: Masterman, Tula, et al. "The landscape of emerging ai agent architectures for reasoning, planning, and tool calling: A survey." arXiv preprint arXiv:2404.11584 (2024).
- エージェントのサーベイ: Wang, Lei, et al. "A survey on large language model based autonomous agents." Frontiers of Computer Science 18.6 (2024): 186345.
- エージェントのビジネス応用: SIERRA, The Guide to AI Agents

なぜAIエージェントを開発するのか

- RPA: Insight Partners, AI Agents are disrupting automation: Current approaches, market solutions and recommendations
- RAG: LlamaIndex, RAG in 2024: advancing to agents
- Text-to-Analytics Agents: Hong, Sirui, et al. "Data interpreter: An LLM agent for data science." arXiv preprint arXiv:2402.18679 (2024).
- エージェント開発の基礎: DeepLearningAI, Functions, Tools and Agents with LangChain

AIエージェントの開発方法とは

- ヘルプデスクエージェント: 電通総研, AIエージェントは何から取り組む？社内取り組み紹介
- エージェントアーキテクチャの参考: Niu, Runliang, et al. “Screenagent: A vision language model-driven computer control agent.” arXiv preprint arXiv:2402.07945 (2024).

AIエージェントの課題とは

- ヘルプデスクエージェント: 電通総研, AIエージェントは何から取り組む？社内取り組み紹介

電通総研

AIエージェントを取り組んでいきましょう！

AITC
AI TRANSFORMATION CENTER トランスフォーメーションセンター