AI Agent

探索智能體技術如何重塑我們的工作

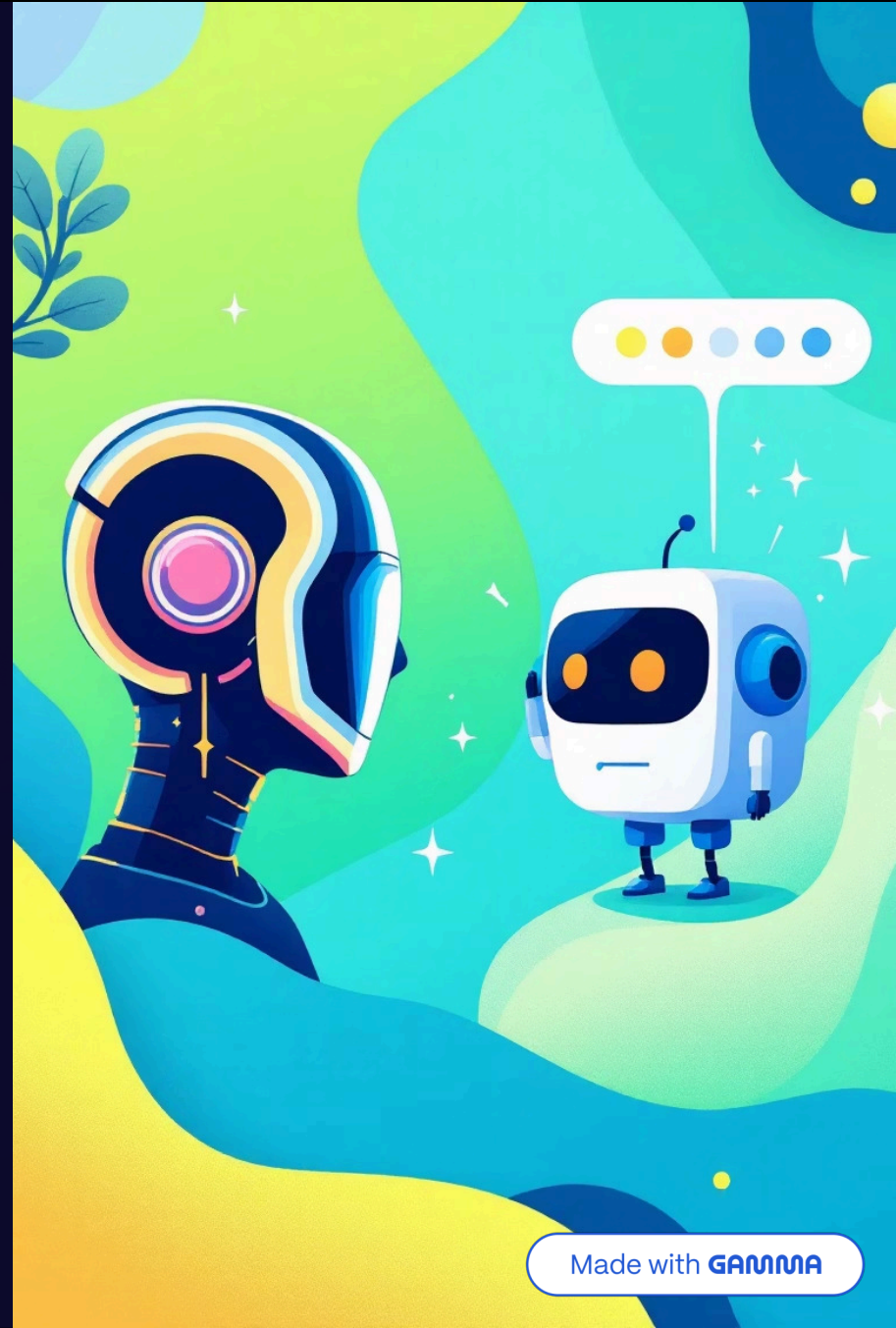
什麼是AI Agent？

智能體定義

能夠**感知環境**、**獨立決策**並**主動執行**動作的人工智能系統

與傳統LLM差異

不只回答問題，更能自主思考和實際執行複雜任務



四大核心能力

感知能力

從文本到多模態演進，GPT-4o具備「眼睛、嘴巴、耳朵」

規劃能力

思維鏈到自主推理，O3模型能完全自主決策

記憶能力

結合上下文擴展與RAG技術，實現長短期記憶

行動能力

從API調用到直接操控電腦，MCP統一工具接口

感知能力的進化歷程

1. **早期LLM**
   僅依賴海量文本數據，接收文本輸入
2. **中介工具**
   利用OCR將圖片、PDF轉換為文本輸入
3. **GPT-4V (2023)**
   初步具備視覺感知，能直接理解圖片信息
4. **GPT-4o (2023底)**
   端到端訓練，理解音訊語調、圖片細節、影片時序

規劃能力的技術突破

思維鏈 (CoT)

讓模型在給出答案前，先拆解問題、分步思考

思維樹 (ToT)

進一步讓模型思考多種思路，選擇最佳方案

多智能體工作流

將任務分解給多個AI模型，各司其職協同工作

自主推理

模型能自主決定何時搜尋、整理、分析，完全由模型控制

記憶能力

1

1. 短期記憶（Working Memory）

作用：維持當前任務的上下文，幫助模型理解「現在正在發生什麼」

技術實現：

- 上下文視窗（Context Window），增加模型的短期記憶
- Agent 在任務執行中產生信息也需被記憶，透過總結存儲、定期回顧形成動態記憶

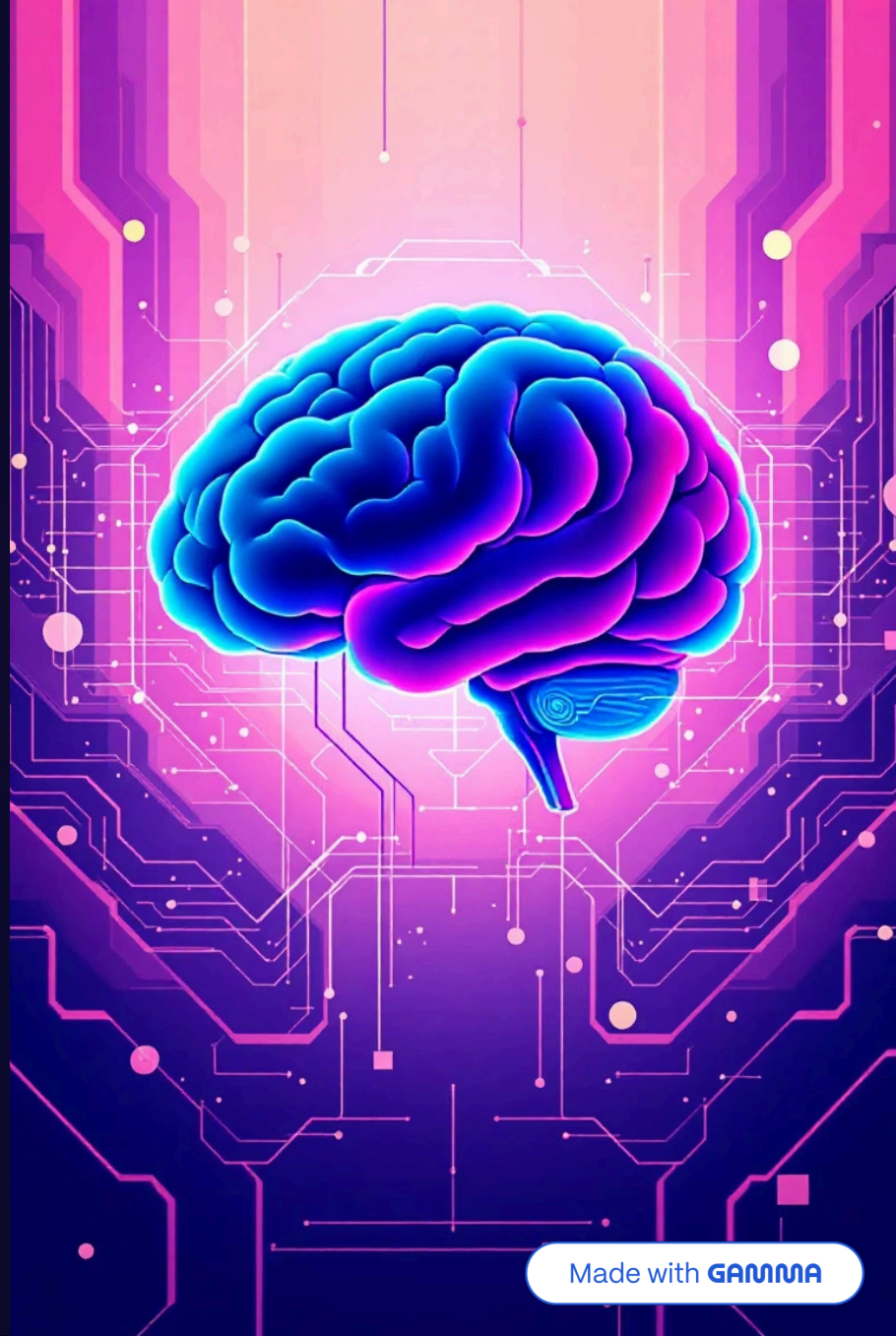
2

2. 長期記憶（Long-Term Memory）

作用：儲存過去的對話、任務紀錄、知識資料，實現資訊持久化

技術實現：

- 檢索增強生成（RAG）：將知識存儲在外部資料庫中，需要時檢索，作為大模型的長期記憶「外掛」，減少幻覺問題



行動能力

- **API 調用 (Function Calling)**：LLM 與外界溝通的最早方式，透過生成 API 調用文本返回結果。大多數 Agent 平台均依賴此方式。
- **直接操控電腦 (Computer Vision)**：訓練大模型從視覺上理解電腦螢幕，進而點擊和操作。初期成功率低，但顯示直接理解像素的能力。
- **瀏覽器控制 (Browser Use)**：網頁自動化工具，間接實現模型控制瀏覽器。Manus 的網頁操作即源於此。
- **模型上下文協議 (MCP)**：統一工具接口標準，降低工具整合門檻，提升模型使用工具效率。

應用領域

IDE程式編寫Agent

GitHub Copilot支持從需求出發，自動編寫、修改程式碼

調查研究 Agent

提供一連串的查詢、比對、總結、輸出。

EDC格式檢查

提供USER直接將XML貼上驗證格式

SPC沒進CHART問題查找

提供USER直接與LLM對話找出問題

*Auto Fix

從BUG發生到LLM修正、編譯、RELEASE、通知USER

AI Agent - Auto Fix

BUG發生
系統傳送問題到AI AGENT

初步審核
Reviewer LLM 執行程式碼的初步審核

工程師審核
工程師人工審核程式碼。若批准，傳送認可到AI AGENT。若退回，將要求AGENT LLM 重修

建置/發布
系統進行編譯，並發布到生產環境

結果通知
通知工程師程式碼發布的最終結果

程式碼生成/修復
AGENT LLM (Builder) 自動修補程式碼

審核報告
生成 Markdown格式的審核報告

觸發編譯
AGENT LLM 觸發程式碼編譯流程

程式碼同步
程式碼同步GIT