Capitolo 1

Gli agenti e il loro mondo

Quello dell'agente non è un concetto nuovo nel machine learning e nell'intelligenza artificiale. Nel *reinforcement learning*, per esempio, la parola *agente* denota un'intelligenza attiva di apprendimento e di decisione. In altre aree, la parola *agente* si allinea più al concetto di un'applicazione o un software automatizzato che fa qualcosa per voi.

In questo capitolo

Definizione di agente

La definizione di agente può essere trovata su qualsiasi dizionario online. Il dizionario inglese *Merriam-Webster Dictionary* (`www.merriam-webster.com/dictionary/agent`) lo definisce nel seguente modo.

- Chi agisce o esercita un potere.
- Qualcosa che produce o può produrre un effetto.
- Un mezzo o uno strumento attraverso il quale un'intelligenza raggiunge un risultato.

La parola *agente*, nel nostro viaggio in questo libro dedicato alla costruzione di agenti, usa quest'ultima definizione. Ciò significa che anche il termine *assistente* sarà sinonimo di *agente*. Anche strumenti come *GPT Assistants* di OpenAI rientrano nella categoria degli agenti a intelligenza artificiale. OpenAI evita di usare la parola *agente* a causa della storia del machine learning, in cui un agente è autonomo e autodecisivo. La Figura 1.1 mostra quattro casi in cui un utente può interagire con un modello LLM (*Large Language Model*) o direttamente oppure tramite un proxy agente/assistente, un agente/assistente o un agente autonomo.

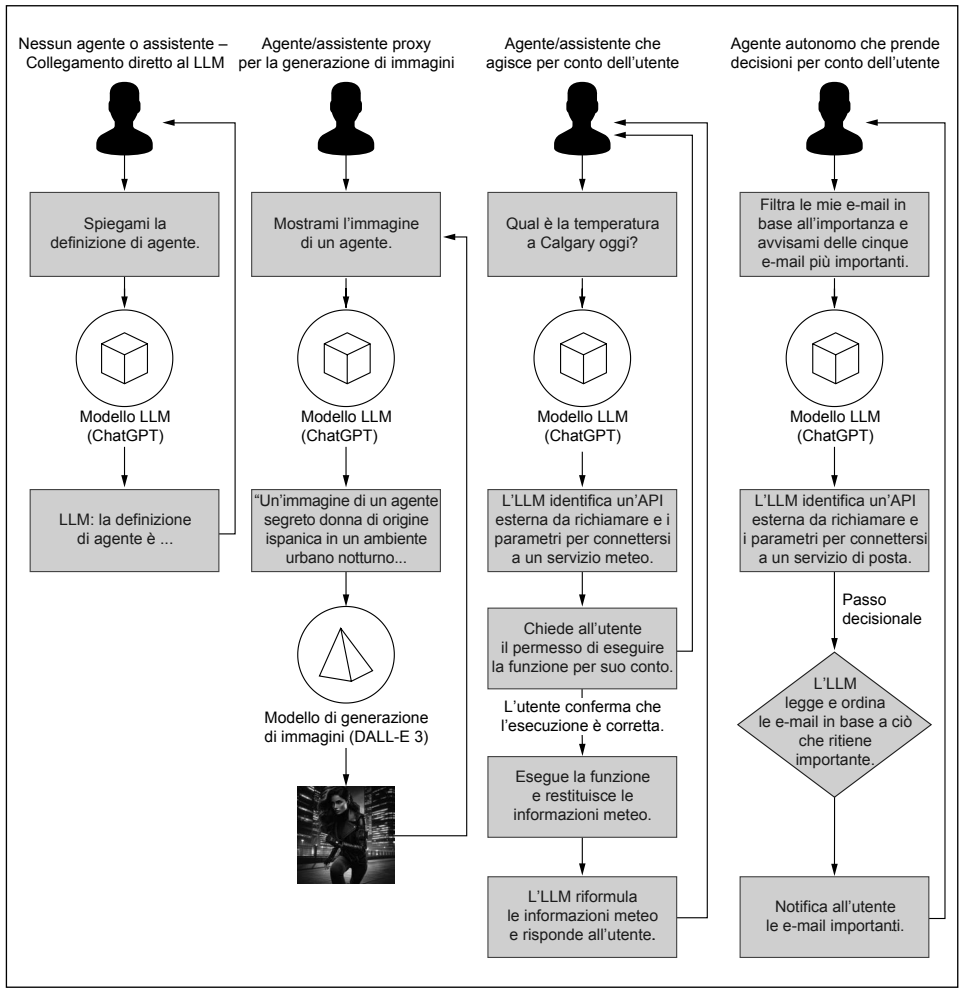

Figura 1.1 Le differenze tra le diverse interazioni con un modello LLM.

Questi quattro casi d'uso sono evidenziati più in dettaglio di seguito.

- *Interazione diretta con l'utente*: se avete utilizzato le versioni precedenti di ChatGPT, avete sperimentato un'interazione diretta con il modello LLM. Non c'è di mezzo alcun agente proxy o alcun altro assistente che interviene per voi.
- *Agente/assistente proxy*: se avete utilizzato *Dall-E 3* tramite ChatGPT, avete sperimentato un'interazione con un agente proxy. In questo caso, un modello LLM si interpone e riformula le vostre richieste in un formato più adatto all'attività. Per esempio, per la generazione di immagini, ChatGPT formula meglio il prompt. Un agente proxy assiste gli utenti ad affrontare attività o modelli non familiari.
- *Agente/assistente*: se avete utilizzato un plugin ChatGPT o un assistente GPT, allora avete sperimentato questo tipo di uso. In questo caso, il modello LLM è a conoscenza delle funzioni del plugin o dell'assistente, e si prepara a effettuare chiamate a questo plugin/funzione. Tuttavia, prima di effettuare una chiamata, il modello LLM richiede

l'approvazione dell'utente. Se approvato, il plugin o la funzione vengono eseguiti e i risultati vengono restituiti al modello LLM, il quale racchiude questa risposta in un testo in linguaggio naturale e la restituisce all'utente.

- *Agente autonomo*: in questo caso, l'agente interpreta la richiesta dell'utente, costruisce un piano e identifica i punti decisionali. Dopodiché, esegue i passaggi del piano e prende in modo indipendente le decisioni richieste. L'agente può richiedere il feedback dell'utente dopo determinate attività fondamentali, ma spesso gli viene data carta bianca per esplorare e imparare, se possibile. Questo tipo di agente solleva preoccupazioni etiche e di sicurezza, come esploreremo più avanti.

La Figura 1.1 illustra i casi d'uso per un singolo flusso di azioni su un modello LLM utilizzando un singolo agente. Per problemi più complessi, spesso dividiamo gli agenti in profili o personaggi (*persona*). A ogni profilo-agente viene assegnato un compito specifico, che esegue impiegando strumenti e conoscenze specializzati.

I *sistemi multi-agente* sono profili per gli agenti che collaborano fra loro in varie configurazioni per risolvere un problema. La Figura 1.2 mostra un esempio di un sistema multi-agente che impiega tre agenti: un controller, o proxy, e due agenti a profilo, controllati dal proxy. Il profilo *coder*, a sinistra, scrive il codice richiesto dall'utente; il profilo *tester*, a destra, è progettato per scrivere unit test. Questi agenti lavorano e comunicano fra loro finché non sono soddisfatti del codice, e poi lo passano all'utente.

La Figura 1.2 mostra una delle, teoricamente infinite, configurazioni ad agenti. Nel Capitolo 4 esploreremo la piattaforma open source di Microsoft, *AutoGen*, che supporta molteplici configurazioni per l'impiego di sistemi multi-agente.

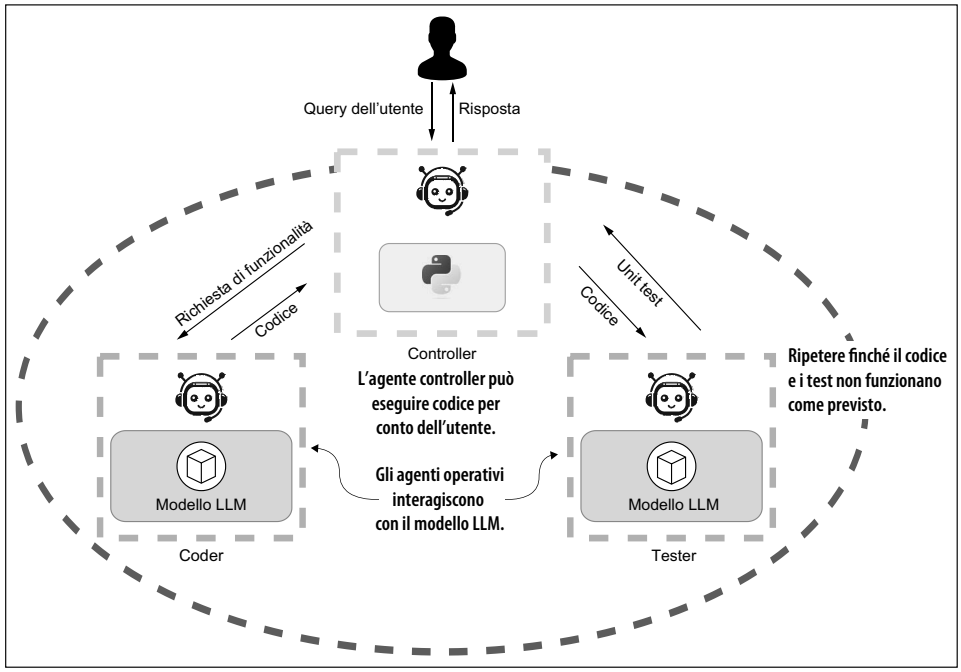

Figura 1.2 In questo esempio di un sistema multi-agente, il controller o proxy comunica direttamente con l'utente. Due agenti, un coder e un tester, lavorano in background per creare il codice e scrivere gli unit test per verificare la correttezza del codice.

I sistemi multi-agente possono funzionare in modo autonomo, ma possono anche funzionare guidati interamente dal feedback umano. I vantaggi dell'utilizzo di più agenti sono simili a quelli di un singolo agente, ma spesso risultano amplificati. Mentre in genere un singolo agente è specializzato in un singolo compito, i sistemi multi-agente possono affrontare più compiti in parallelo. Più agenti possono anche fornire feedback e valutazioni, riducendo gli errori durante l'esecuzione dei compiti.
Come possiamo vedere, un agente o un sistema ad agenti a intelligenza artificiale può essere assemblato in più modi. Tuttavia, un agente può anche essere assemblato utilizzando più componenti. Nel prossimo paragrafo, tratteremo argomenti che vanno dal profilo di un agente alle azioni che esso può eseguire, nonché alla memoria e al planning.

I sistemi a componenti di un agente

Gli agenti possono essere unità complesse, composte da sistemi multi-componente. Questi componenti sono gli strumenti che l'agente impiega per completare il proprio obiettivo o i compiti che gli sono stati assegnati, ma può anche crearne di nuovi. I componenti possono essere sistemi semplici o complessi, in genere suddivisi in cinque categorie.
La Figura 1.3 descrive le principali categorie di componenti che possono essere presenti in un sistema mono-agente. Ogni elemento avrà dei sottotipi, che possono definire il tipo, la struttura e l'uso del componente. Al centro di tutti gli agenti c'è il profilo e il personaggio (*persona*); da ciò si estendono i sistemi e le funzioni che potenziano l'agente.

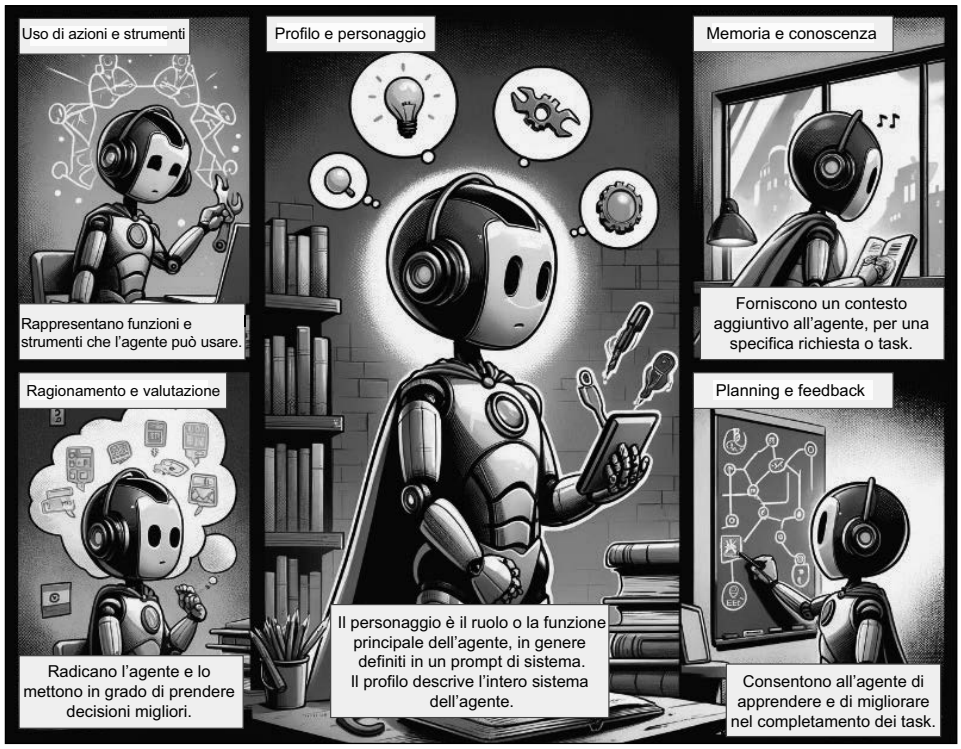

Figura 1.3 I cinque componenti principali di un sistema mono-agente (immagine generata tramite DALL-E 3).

Il profilo e il personaggio (*persona*) dell'agente, mostrati nella Figura 1.4 rappresentano la descrizione di base dell'agente. Il personaggio, *persona* in inglese e spesso chiamata *prompt di sistema*– guida un agente nel completare i task, nell'imparare a rispondere e in altri dettagli. Include elementi come il background (per esempio, essere un programmatore o uno scrittore) e dati demografici, e può essere generata dall'utente, con l'assistenza di un modello LLM o con tecniche basate sui dati, inclusi gli algoritmi evolutivi.

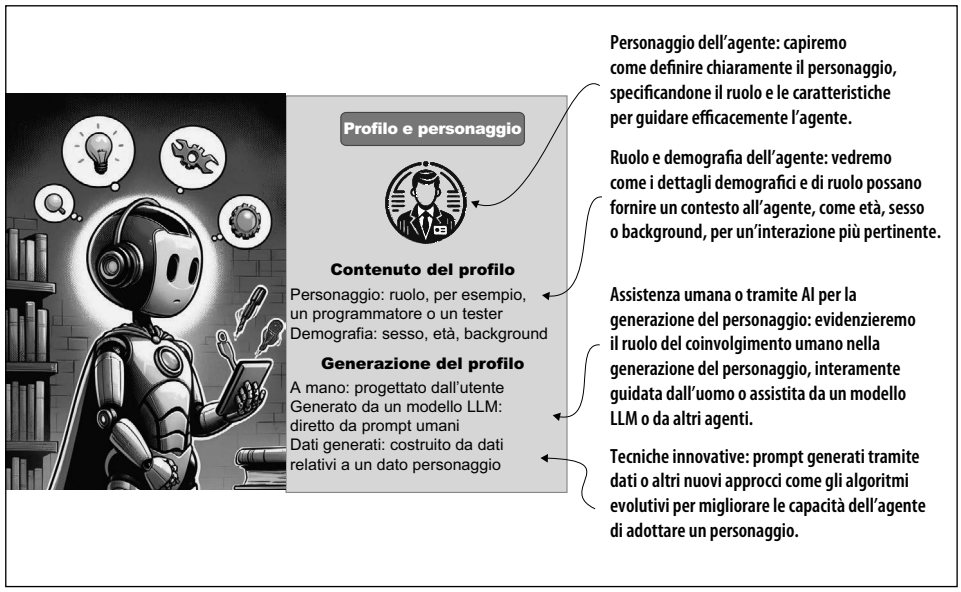

Figura 1.4 Uno sguardo approfondito sul modo in cui esploreremo la creazione di profili per gli agenti.

Vedremo come creare profili/personaggi efficaci e specifici per gli agenti, attraverso tecniche come le griglie di valutazione e la pertinenza. Inoltre, spiegheremo gli aspetti dei profili formulati dall'utente rispetto a quelli formulati da un'intelligenza artificiale (da un modello LLM), comprese tecniche innovative che creano profili utilizzando dati e algoritmi evolutivi.

NOTA

Il profilo dell'agente o dell'assistente è composto da più elementi, tra cui il personaggio. Può essere utile immaginare che i profili descrivano il lavoro che l'agente/assistente deve svolgere e gli strumenti di cui ha bisogno.

La Figura 1.5 illustra le azioni dei componenti e l'uso degli strumenti nel contesto degli agenti che coinvolgono attività volte al completamento di un task o all'acquisizione di informazioni. Queste azioni possono essere categorizzate in completamento di un task, esplorazione e comunicazione, con diversi livelli di effetti sull'ambiente e sugli stati interni dell'agente. Le azioni possono essere generate manualmente, tramite il recupero dei ricordi, o seguendo piani predefiniti, che influenzano il comportamento dell'agente e ne migliorano l'apprendimento.

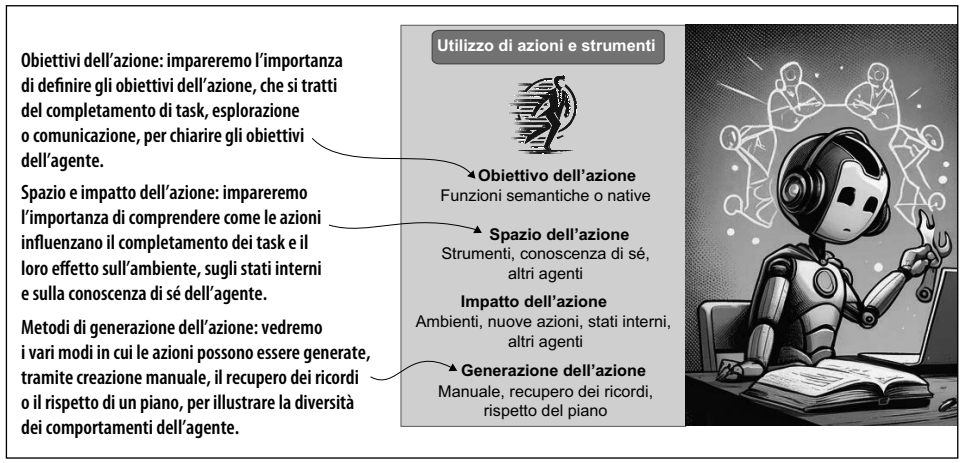

Figura 1.5 Gli aspetti delle azioni-agente che esploreremo in questo libro.

Comprendere l'obiettivo dell'azione ci aiuta a definire obiettivi chiari per il completamento dei task, l'esplorazione o la comunicazione. Riconoscere l'effetto dell'azione rivela come le azioni influenzano i risultati del task, l'ambiente dell'agente e i suoi stati interni, contribuendo a ottenere un processo decisionale efficiente. Infine, comprendere i metodi di generazione dell'azione ci dota della conoscenza necessaria per creare le azioni manualmente, per richiamarle dalla memoria o per seguire piani predefiniti, migliorando la nostra capacità di modellare efficacemente il comportamento dell'agente e i processi di apprendimento.

La Figura 1.6 mostra in modo più dettagliato il componente memoria e conoscenza. Gli agenti usano la memoria e la conoscenza per dotare il contesto di informazioni più pertinenti, limitando al contempo il numero di token utilizzati. Le strutture di memoria e conoscenza possono essere unificate, dove entrambi i sottoinsiemi seguono una struttura singola o ibrida, che coinvolge un mix di diverse forme di recupero dei contenuti. I formati della memoria e della conoscenza possono variare ampiamente: linguaggi (per esempio, documenti PDF), database (relazionali, a oggetti o documenti) ed embedding, che semplificano la ricerca per similitudine semantica tramite rappresentazioni vettoriali, fino a semplici liste, che fungono da memoria degli agenti.

La Figura 1.7 mostra il componente di ragionamento e valutazione di un sistema ad agenti. La ricerca e le applicazioni pratiche hanno dimostrato che i modelli LLM e gli agenti possono ragionare in modo efficace. I sistemi di ragionamento e valutazione arricchiscono il flusso di lavoro di un agente fornendogli la capacità di riflettere sui problemi e di valutare le soluzioni.

La Figura 1.8 mostra il componente di planning/feedback dell'agente, e il suo ruolo nell'organizzazione dei task per raggiungere obiettivi di livello superiore. Può essere suddiviso nei due approcci seguenti.

- *Planning senza feedback*: gli agenti autonomi prendono decisioni in modo indipendente.
- *Planning con feedback*: il monitoraggio e la modifica dei piani si basano su varie fonti di input, tra cui i cambiamenti ambientali e il feedback umano diretto.

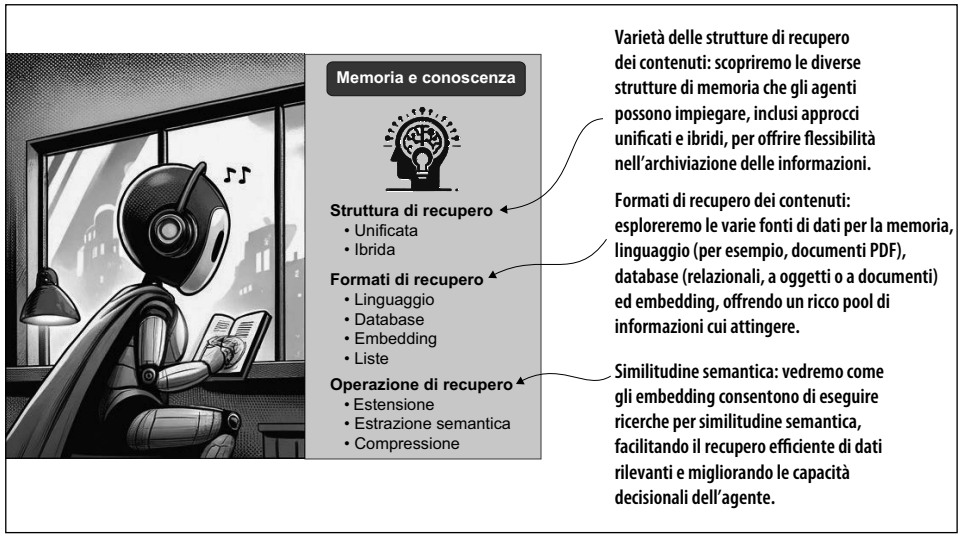

Figura 1.6 Il ruolo e l'uso della memoria e della conoscenza da parte dell'agente.

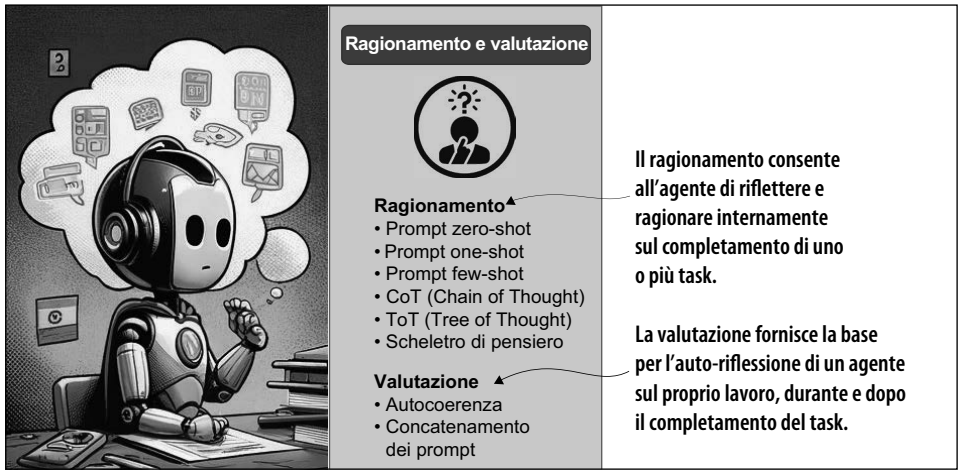

Figura 1.7 Il componente di ragionamento e valutazione.

Nell'ambito del planning, gli agenti possono impiegare il *ragionamento single-path*, il *ragionamento sequenziale* attraverso ogni fase di un'attività o il *ragionamento multi-path*, per esplorare più strategie e salvare quelle più efficienti per un uso futuro. Anche i planner esterni, che possono essere costituiti da codice o altri agenti, possono svolgere un ruolo nella gestione dei piani.

Ognuno dei tipi di agenti, agente/assistente proxy, agente/assistente o agente autonomo, può utilizzare alcuni o tutti questi componenti. Anche il componente di planning ha un suo ruolo, oltre a quello di agente autonomo, e può potenziare efficacemente anche un normale agente.

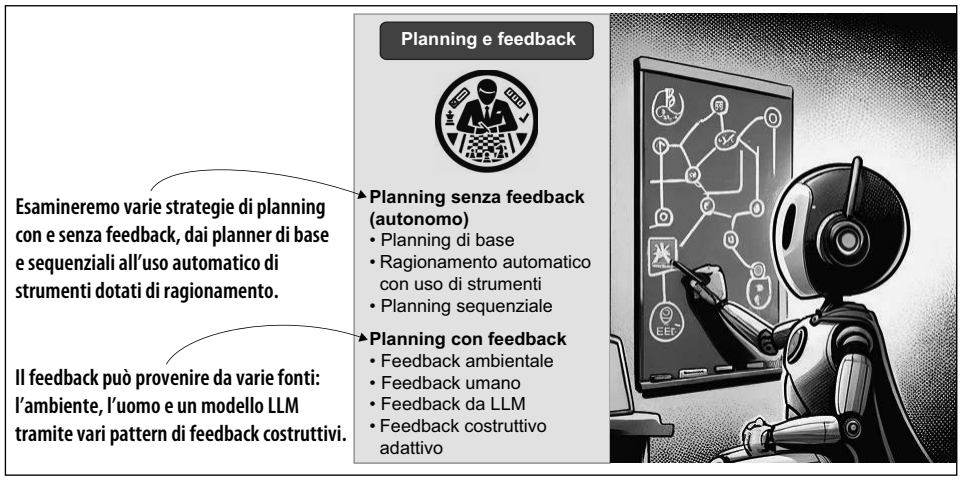

Figura 1.8 Il ruolo del planning e del ragionamento dell'agente.

L'era degli agenti

Gli agenti e gli assistenti a intelligenza artificiale sono passati rapidamente dall'apice della ricerca allo sviluppo software mainstream. L'elenco degli strumenti e delle piattaforme che aiutano nella costruzione e nel potenziamento degli agenti cresce continuamente. A un osservatore esterno, tutto ciò potrebbe sembrare un gran clamore volto solo a gonfiare il valore di una tecnologia interessante ma sopravvalutata.

Nei mesi successivi al rilascio iniziale di ChatGPT, è nata una nuova disciplina chiamata *prompt engineering*: gli utenti hanno scoperto che l'impiego di determinati pattern e tecniche di composizione dei prompt consentiva loro di generare output migliori e più coerenti. Tuttavia, gli utenti hanno anche capito che il prompt engineering aveva dei limiti: aiutava, sì, ma solo fino a un certo punto.

Il prompt engineering è ancora oggi un modo eccellente per interagire direttamente con i modelli LLM, come ChatGPT. Nel tempo, molti utenti hanno scoperto che un prompting efficace richiedeva iterazioni, riflessioni e nuove iterazioni. Da queste scoperte sono emersi i primi sistemi ad agenti, come AutoGPT, che hanno catturato l'attenzione della community.

La Figura 1.9 mostra la struttura originaria di AutoGPT, forse il primo sistema ad agenti autonomo. L'agente è progettato per iterare una sequenza pianificata di task, che definisce esaminando l'obiettivo dell'utente. A ogni iterazione di passaggi del task, l'agente valuta l'obiettivo e determina se il task è concluso. Se il task non è concluso, l'agente può ripianificare i passaggi e aggiornare il piano in base a nuove conoscenze o a un feedback umano.

AutoGPT è stato il primo a dimostrare la potenza dell'utilizzo del planning dei task e dell'iterazione con i modelli LLM. Da qui, e in parallelo, nella community sono esplosi altri sistemi e framework ad agenti, utilizzando sistemi simili di planning e iterazione dei task. È generalmente accettato che il planning, l'iterazione e la ripetizione siano i migliori processi per risolvere obiettivi complessi e multiforme per un modello LLM.

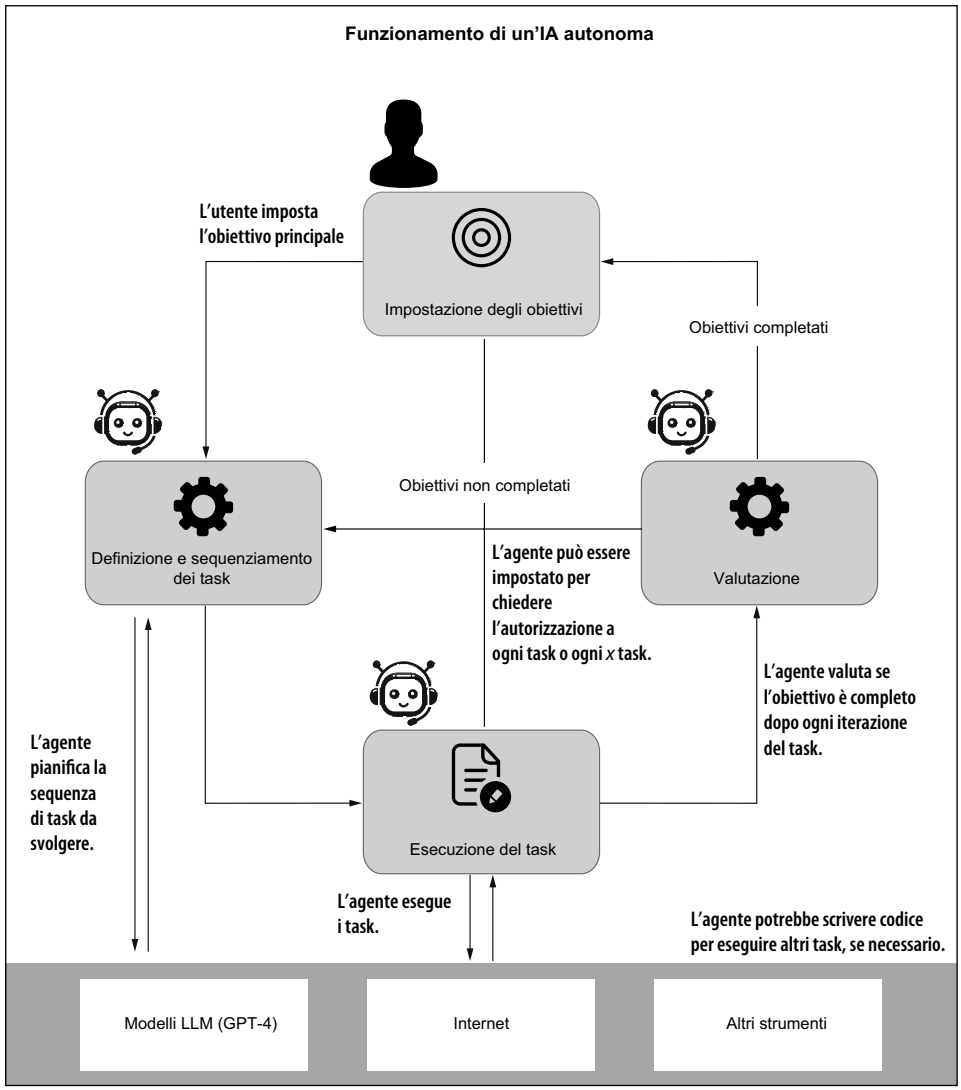

Figura 1.9 Il progetto originale del sistema ad agenti AutoGPT.

Tuttavia, i sistemi ad agenti autonomi richiedono una fiducia nel processo decisionale dell'agente, nel sistema di controllo/valutazione e nella definizione dell'obiettivo. La fiducia è qualcosa che si acquisisce nel tempo. La nostra mancanza di fiducia deriva dalla nostra mancanza di comprensione delle capacità di un agente autonomo.

NOTA

L'intelligenza artificiale generale (AGI) è una forma di intelligenza che può imparare a svolgere qualsiasi compito tipico di un essere umano. Molti professionisti, in questo nuovo mondo dell'intelligenza artificiale, sono convinti che un'AGI che utilizzi sistemi ad agenti autonomi sia un obiettivo raggiungibile.

Per questo motivo, molti degli strumenti ad agenti mainstream e in produzione non sono autonomi. Tuttavia, forniscono comunque un vantaggio significativo nella gestione e nell'automazione delle attività tramite GPT (ovvero tramite modelli LLM). Pertanto, poiché il nostro obiettivo in questo libro sarà quello di trattare tutte le forme di agenti, molte applicazioni pratiche saranno guidate da agenti non autonomi.
Gli agenti e strumenti ad agenti sono solo lo strato più popolare di un nuovo paradigma di sviluppo di applicazioni. Nel prossimo paragrafo esploreremo questo nuovo paradigma.

L'interfaccia a intelligenza artificiale

Il paradigma dell'agente a intelligenza artificiale non rappresenta solo un cambiamento nel modo in cui utilizziamo i modelli LLM, ma è percepito anche come un cambiamento nel modo in cui sviluppiamo il software e gestiamo i dati. Il software e i dati non si interfacceranno più tramite interfacce utente, API (*Application Programming Interface*) e linguaggi specializzati per query, come SQL. Saranno invece progettati per interfacciarsi in linguaggio naturale.
La Figura 1.10 mostra un'istantanea di alto livello di quale potrebbe essere l'aspetto questa nuova architettura, e del ruolo degli agenti a intelligenza artificiale. Dati, software e applicazioni cambiano per supportare nuove interfacce semantiche in linguaggio naturale. Queste interfacce a intelligenza artificiale consentono agli agenti di raccogliere i dati e di interagire con le applicazioni, ma anche con altri agenti o con applicazioni ad agenti. Ciò rappresenta un cambiamento radicale nel modo in cui interagiamo con il software e le applicazioni.
Un'*interfaccia a intelligenza artificiale* è una raccolta di funzioni, strumenti e livelli di dati che espongono dati e applicazioni tramite il linguaggio naturale. In passato, la parola *semantica* è stata ampiamente utilizzata per descrivere queste interfacce e perfino alcuni strumenti usano questo nome; tuttavia, l'aggettivo "semantico" può avere diversi significati e utilizzi. Pertanto, in questo libro, userò il termine *interfaccia a intelligenza artificiale*.
La costruzione di interfacce a intelligenza artificiale causerà un avanzamento degli agenti che hanno bisogno di impiegare servizi, strumenti e dati. Con questa nuova potenza arriverà una maggiore accuratezza nel completamento dei task e verranno realizzate applicazioni più affidabili e autonome. Anche se un'interfaccia a intelligenza artificiale potrebbe non essere appropriata per tutti i tipi di software e di dati, si estenderà a molti casi d'uso.

Il panorama degli agenti

Gli agenti GPT rappresentano un cambiamento radicale nel modo in cui i consumatori e gli sviluppatori affrontano un po' tutto, dalla ricerca di informazioni alla creazione di software, all'accesso ai dati. Quasi ogni giorno, su GitHub o in un documento di ricerca spunta un nuovo framework, componente o interfaccia ad agenti. La cosa può essere fin eccessiva e anche preoccupante per il nuovo utente che cerchi di comprendere che cosa sono i sistemi ad agenti e come utilizzarli.

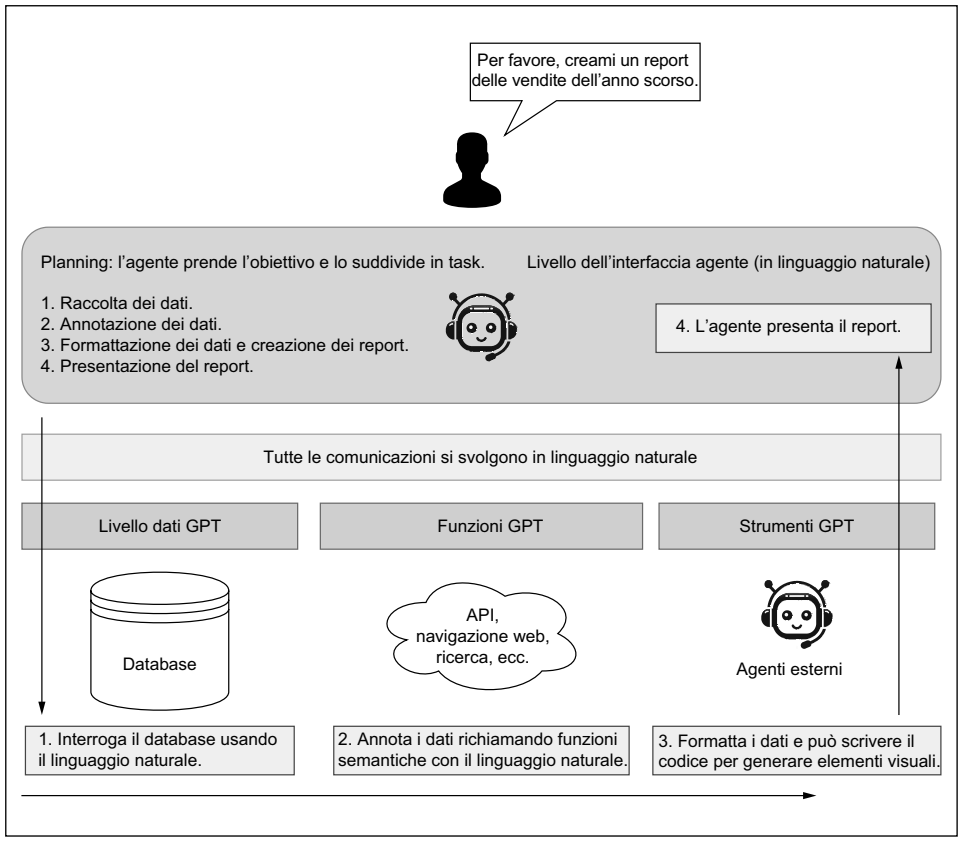

Figura 1.10 Una visione delle interazioni fra agenti e sistemi software.

Riepilogo

- Un agente è un'entità che agisce o esercita un potere, produce un effetto o funge da mezzo per ottenere un risultato. Un agente automatizza l'interazione con un modello LLM tramite l'intelligenza artificiale.
- Il termine assistente è sinonimo di agente. Entrambi i termini comprendono strumenti come *GPT Assistants* di OpenAI.
- Gli agenti autonomi sono in grado prendere decisioni indipendenti, e la loro distinzione rispetto agli agenti non autonomi è fondamentale.
- I quattro tipi principali di interazioni con i modelli LLM includono l'interazione diretta dell'utente, i proxy agente/assistente, gli agenti/assistenti e gli agenti autonomi.
- I sistemi multi-agente prevedono che i profili-agente collaborino, spesso controllati da un proxy, per portare a termine compiti complessi.
- I componenti principali di un agente includono il profilo/personaggio (*persona*, in inglese), le azioni, la conoscenza/memoria, il ragionamento/valutazione e il planning/feedback.

- I profili e i personaggi degli agenti guidano i compiti, le risposte e altri dettagli, spesso includendo informazioni di background e dati demografici.
- Le azioni e gli strumenti per gli agenti possono essere generati manualmente, richiamati dalla memoria o seguire piani predefiniti.
- Gli agenti utilizzano strutture di memoria e conoscenze per ottimizzare il contesto e ridurre al minimo l'utilizzo di token tramite vari formati, che vanno dai documenti agli embedding.
- I sistemi di ragionamento e valutazione consentono agli agenti di riflettere sui problemi e di valutare le soluzioni utilizzando pattern di prompting quali zero-shot, one-shot e few-shot.
- I componenti di planning/feedback organizzano i task per raggiungere gli obiettivi utilizzando ragionamenti single-path o multi-path e integrando i feedback dall'ambiente e umano.
- L'ascesa degli agenti a intelligenza artificiale ha introdotto un nuovo paradigma di sviluppo software, facendo passare dalle interfacce tradizionali a quelle basate sul linguaggio naturale.
- Comprendere la progressione e l'interazione di questi strumenti aiuta a sviluppare sistemi ad agenti, siano essi singoli, multipli o autonomi.