

大型語言模型提示技術的實證分類：實務者指南

Oluwole Fagbohun
Tech Team
Changeblock
倫敦，SW1H 0HW，英國
oluwole.fagbohun@changeblock.com

Rachel M. Harrison
GenAI Lab
Ophiuchus LLC
多佛，DE 19904，美國
rae@ophiuchus.ai

Anton Dereventsov
GenAI Lab
Ophiuchus LLC
Dover, DE 19904, USA
anton@ophiuchus.ai

摘要

由於大型語言模型（LLMs）開發的快速進展，使用提示（prompts）來程式化這些模型近期獲得了顯著關注。然而，繁多的提示工程技術使得想要利用這些工具的實務者面臨一個令人眼花撩亂的領域。為了更有效率且有成效地使用 LLMs，彙整一份完整的提示技術清單並建立一套標準化、跨領域的分類框架相當重要。在本調查中，我們從學術與實務兩個觀點檢視一些最知名的提示技術，並將它們分類為七個不同類別。我們針對每個類別提供概述，旨在釐清其獨特貢獻，並以實際案例展示其實務應用，為實務者提供一個能依特定領域理解與分類提示技術的結構化框架。我們相信此方法將有助於簡化複雜的提示工程領域，並促進 LLMs 在各種應用中的更有效運用。透過為實務者提供系統化的提示分類方法，我們旨在協助其掌握為對話式預訓練 LLMs 設計有效提示的複雜性，並激發其各自領域中的新可能性。

關鍵詞 large language models . LLMs . prompt engineering . prompt design . prompting techniques . ChatGPT

1 引言

大型語言模型（LLMs）(Radford et al., 2019) 近來人氣暴增，從根本上重塑了機器學習的版圖 (Zhao et al., 2023)。早期模型 (Zhou et al., 2023; Bommasani et al., 2021) 以範圍與功能受限為特徵，主要針對文字分類與情感分析等特定任務而設計；隨著神經網路與深度學習技術的引入，自然語言處理（NLP）領域發生了轉變，模型得以從龐大資料集學習，並在更廣泛的任務上展現適應性 (Petroni et al., 2019; Brown et al., 2020)。在這段演化歷程中，一個關鍵時刻是 transformer 架構的誕生 (Vaswani et al., 2017)。該架構的可擴展性與處理長距離相依關係的卓越能力，為開發更複雜且更強大的模型奠定了基礎。隨後，像 BERT (Devlin et al., 2018) 和 GPT (Radford et al., 2019) 這類模型的引入展示了先在大量資料上進行預訓練，接著再微調以在特定任務上表現出色的變革性潛力。

1.1 會話式預訓練大型語言模型

在這個快速演進的領域中，一類被稱為會話式僅解碼器 transformer 變體 (Vaswani et al., 2017) 的 LLMs，如 GPT (Brown et al., 2020; Ouyang et al., 2022)、LaMDA (Thoppilan et al., 2022),

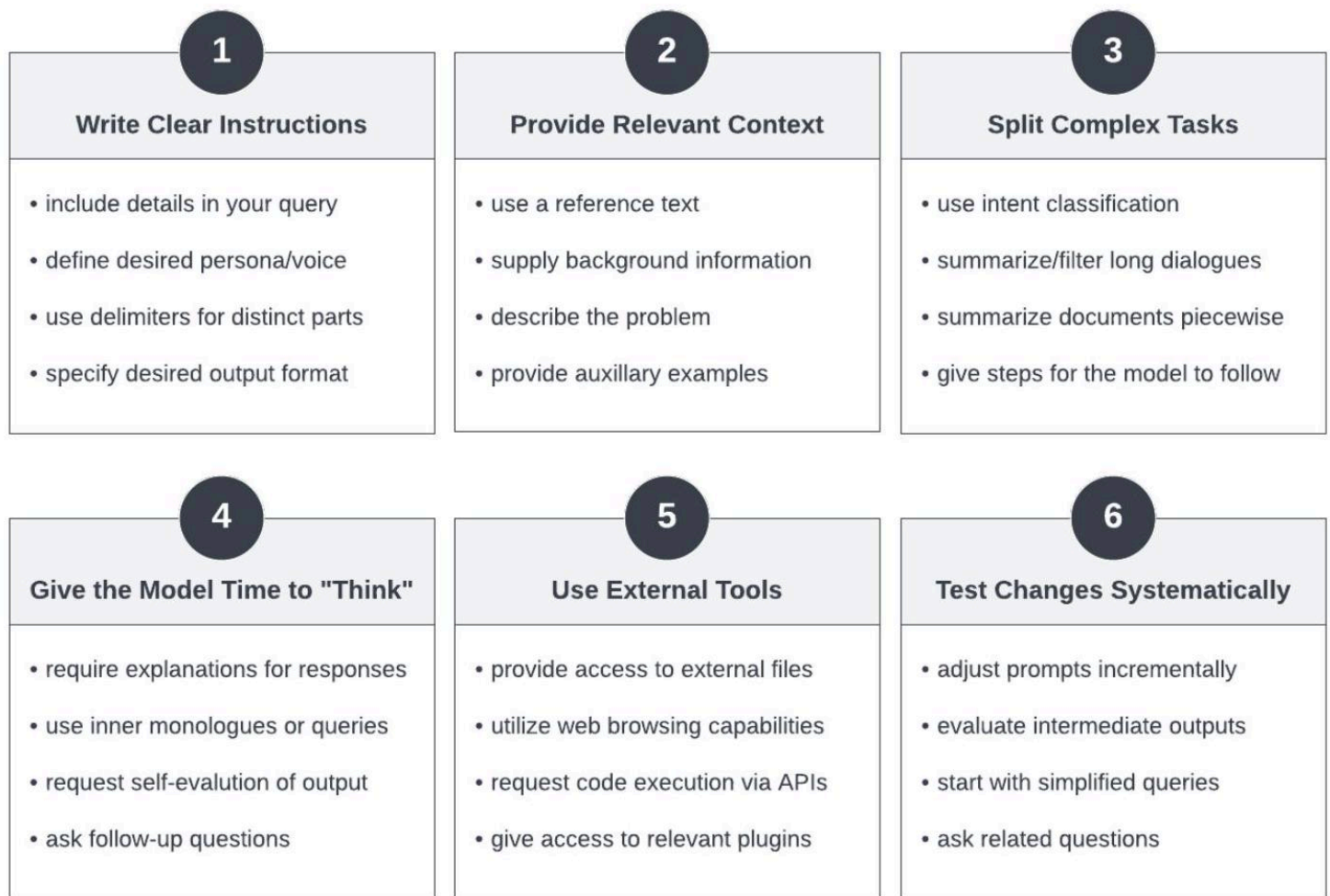


圖 1：圖 1：有效提示設計之常見最佳實踐範例。

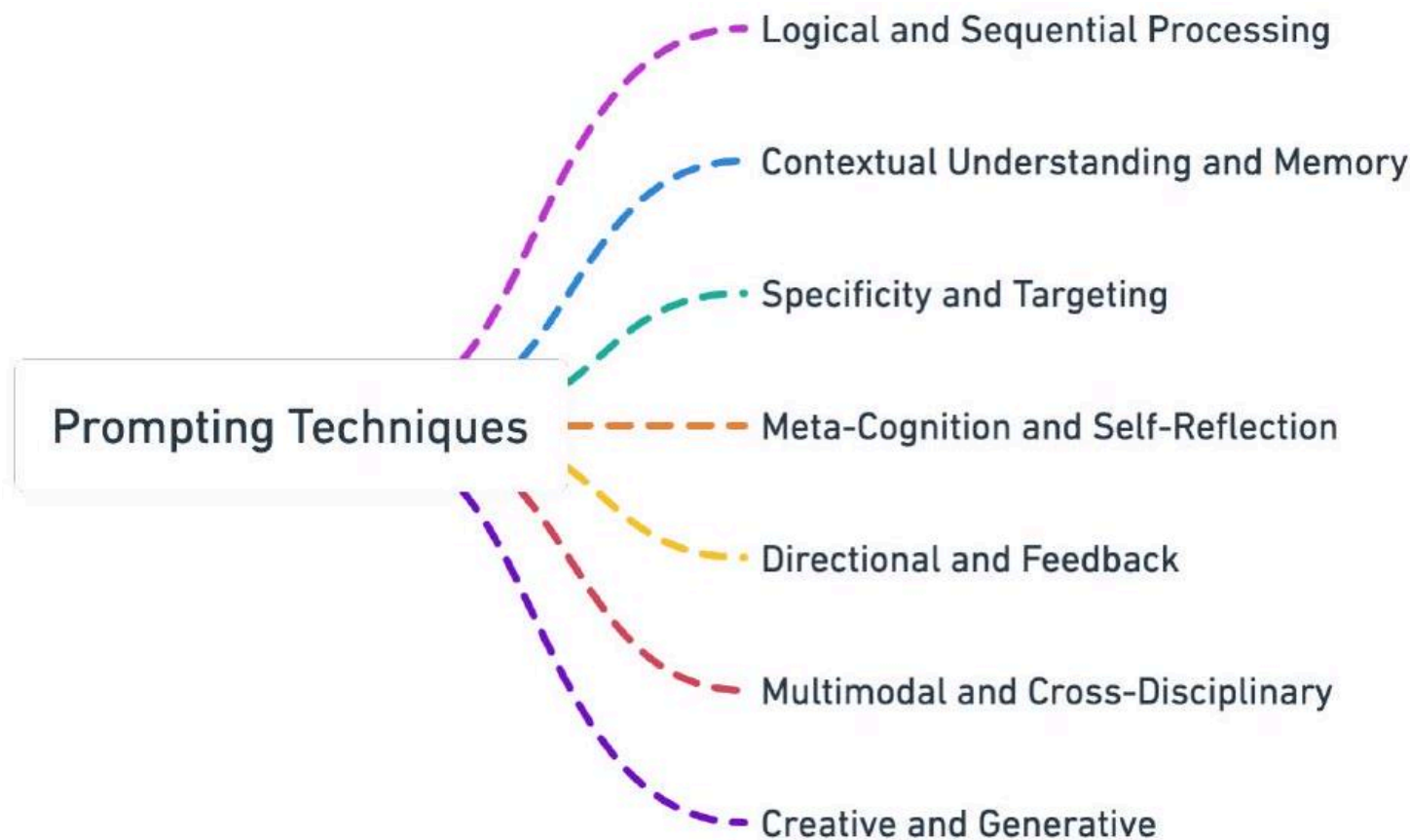
PaLM (Chowdhery et al., 2023)、LLaMa (Touvron et al., 2023) 與 Mistral (Jiang et al., 2023) 已成為改寫自然語言處理領域的關鍵角色。這些模型具備卓越的理解、生成與詮釋人類語言的能力，對各種領域產生深遠影響，從以科技為主的金融 (Wu et al., 2023; Liu et al., 2023) 與程式設計 (Xu et al., 2022; Roziere et al., 2023)，到以人為本的教育 (Kasneci et al., 2023; Kung et al., 2023) 與醫療保健 (Abbasian et al., 2023; Thirunavukarasu et al., 2023)。此外，由於它們的對話性質以及像 ChatGPT (Brown et al., 2020; Yang et al., 2023) 這類以聊天為主的介面廣受歡迎，這些模型的複雜運算能力幾乎對任何有網路連線和對人工智慧 (AI) 有基本興趣的人都變得易於使用。然而，為了讓模型正確回應使用者的查詢，使用者必須先知道如何透過構建適當的 prompt 與模型溝通。

1.2 Prompting for LLMs

提示詞是人類溝通與 LLMs 計算能力之間的翻譯橋樑。它由自然語言指令構成，作為中介語言，將人類的請求翻譯成機器可執行的任務 (Liu et al., 2023)。除了簡單的指示之外，提示詞還為互動建立情境框架，傳達資訊的重要性並定義模型輸出所需的格式與內容。

這些指令可以從像問答這樣的直接查詢到需要提供情境輸入、並對輸出格式提出具體要求的更複雜任務。

隨著 LLMs 持續進化，提示 (prompts) 的範疇也隨之擴大，催生出能推動這些模型能力邊界的新型互動範式。隨著像 ChatGPT (Bang et al., 2023) 這類更高級 LLMs 的出現，提示旨在處理的任務複雜度也隨之提高。雖然以自然語言進行溝通的提示渠道看似廣泛可及，但研究顯示，非 AI 專家與主要採取機會式提示策略的個體，在利用 LLMs 解決指定任務方面普遍效率低落 (Zamfirescu-Pereira et al., 2023)。因此，這個快速演變的環境已將焦點轉向優化提示的策略性使用，並催生出對專職提示專家的需求。



\captionsetup{labelformat=empty}
圖 2：提示技術分類概覽。

1.3 提示工程

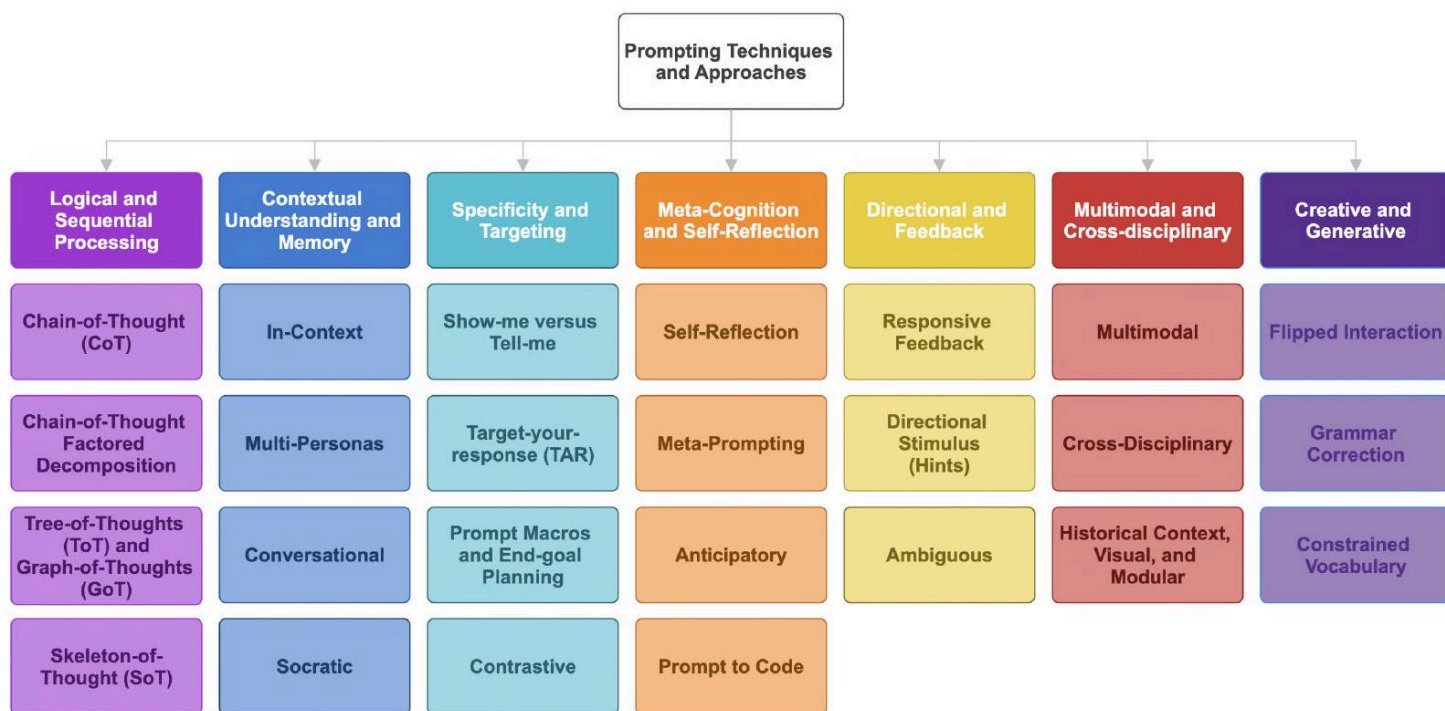
提示工程是設計輸入（提示詞）以引導 LLMs 產生特定且相關輸出的策略性過程。這類似於提出正確的問題或指令，以確保獲得期望的答案。提示工程師運用 AI 模型的預測能力與大量訓練資料，並結合人類溝通的細微差異，策略性地影響模型行為，而無需進行針對特定任務的訓練（Brown et al., 2020；Petroni et al., 2019；Radford et al., 2019；Schick and Schütze, 2021）。有效的提示工程會戰術性地採用各種提示設計的最佳實務（參見圖 1，來源於 OpenAI¹），以最大化 LLMs 的效能，使其在特定任務上更準確且更相關。這不僅涉及提示詞的創建，還包括根據回饋和期望結果對其進行測試與精進。

此外，隨著要求 LLMs 執行的任務日益複雜，有效的提示設計常常需要以嶄新且錯綜的方式結合多個提示。而由於這些模型採用自然語言做為程式化介面，使它們對變化極為敏感，即使在提示的措辭、結構或語境上做出微小修改，也可能導致 LLMs 產生顯著不同的結果（Sclar et al., 2023；Dong et al., 2023）。提示工程師必須具備廣泛的知識基礎，不僅要了解 LLMs 的運作原理，還要理解每一種特定提示模式如何融入整體提示技術的框架，以及在結合使用時彼此如何互動、改進與相互影響。這需要對工程師可能採用的每種模式的優缺點有細緻的理解，因為提示的品質與結構會直接影響 LLMs 如何處理資訊並產生輸出。雖然精心設計的提示（prompts）能引導這些模型生成更準確、相關且符合上下文的回應，但設計不良的提示可能導致模糊、不相關甚至錯誤的輸出（Brown et al., 2020；Ouyang et al., 2022）。因此，LLMs 在實際應用中的效能在很大程度上取決於提示工程的品質，以及提示工程師的專業能力。

幸好，研究文獻提供了寶貴資源，透過呈現多種適用於各類情境的良好記錄提示範式，協助進行適當的提示設計。然而，提示技術種類繁多、表現形式多樣且來源分散，常使人難以回想或檢索這些豐富資訊。因此，有必要對提示技術進行結構化的分類。這類分類不僅能協助實務者做出適切的提示選擇，也能突顯正確提示設計在成功運用 LLMs 的關鍵重要性，確保在日益擴展的應用場景中，LLMs 能有效且高效率地產出期望的結果。

1.4 分類框架

在本次綜述中，我們提出了一個全面的框架，用以分類近年學術文獻中針對會話式預訓練大型語言模型（LLMs）所提出的各類提示技術（見圖 2）。我們的分類方法建立在跨學科的基礎上，認知到 LLMs 的應用橫跨多個不同領域與學科。我們將這些多樣的技術與方法劃分為七個不同的類別，每一類別代表對其方法與預期用途的高階概念化。透過將這些方法組織成具內在關聯的群組，我們為實務工作者提供了選擇最適提示策略的清晰路線圖。



\captionsetup{labelformat=empty}
圖3：圖3：提示技術與方法的分類。

針對其特定應用。不論實務工作者是要生成創意內容、回答問題，或進行自然語言對話，我們的框架都能讓他們輕鬆辨識最相關的分類，並進一步探索與其特定目標相符的技術。

為了進一步提升本次調查的可用性，我們在每種可適用的方法後都參考了相關研究，以便對最感興趣的方法進行更多探索。此外，我們為所有提示方法提供了實際範例，說明這些技術如何在實務中有效運用。透過此一方式，我們不僅讓實務者掌握分類知識，還提供實作這些技術所需的實務見解。

圖 3 所示的提示技術分類如下：

邏輯與序列處理：將複雜推理任務拆解為可管理步驟的技術，改善問題解決能力，並使 LLMs 展現更具科學性與類人化的推理。

情境理解與記憶：用於回憶與參照過去互動的技術，以提供無縫的對話體驗並在長篇對話中確保一致性與相關性。

具體性與目標導向：透過區分資訊類型並與特定目標和目的對齊的技術，以提升 LLMs 的精確度並鼓勵以目標為導向的回應。

元認知與自我反思：透過內省式提示、預判需求與產生程式碼等技術，提升 LLMs 的輔助能力。

導向與回饋：用以引導模型執行特定任務或提供可幫助模型隨時間改善回應的回饋的技術。

多模態與跨領域：整合多元輸入模式與知識領域以提升通用性，並將 LLM 應用擴展到情境複雜的場景的技術。

創意與生成：推動 LLM 產出具創意、引人入勝與探索性內容的技術，適用於說故事、教育與創意寫作等任務。

2 提示技術與方法的分類

在本節中，我們提出一個結構化框架，用以分類在大型語言模型領域中使用多樣提示技術。結合跨學科研究的見解，我們已將這些技術系統性地組織為七個不同類別，以便提供學術文獻中提示設計現狀的全面概覽。鑑於 LLM 在各學科的廣泛應用，

此框架為實務者與研究者提供一個結構化的視角，以便在提示設計的錯綜複雜中航行。其主要目的在於促進對可用方法的深入理解，同時作為選擇與實作這些技術以達成特定目標的實用指南。透過納入相關研究與實際範例的參考，我們的分類旨在使個人不僅能掌握有效提示工程的細微差別，還能在實務情境中有效運用這些技術，進而提升 LLMs 在自然語言處理任務中的實用性。

在發揮 LLMs 潛力時，有效的提示工程至關重要。提示的設計會大幅影響這些模型的行為，因此在選擇提示時需要有結構化且有意圖的方法。此框架可作為一個寶貴資源，協助研究人員和實務者在為各種應用設計提示時做出明智選擇。它鼓勵對任務特定需求進行深入思考，使使用者能夠針對性地調整提示以達到最佳的 LLMs 結果，同時促進在涉及會話式預訓練 LLMs 的研究與應用工作中實現透明性與可重現性。

2.1 邏輯與序列處理

邏輯與序列處理技術允許 LLM 透過將複雜的多步推理任務拆解成可管理的步驟來處理它們。屬於此類別的方法基於將任務拆分為較小組件的理念，並在其應用背後具有科學推理與數學依據。這些方法不僅增強模型的問題解決能力，還提供了一個更直觀且類人推理的框架，為 LLM 在各領域的應用開啟新途徑。欲了解邏輯與序列處理技術的較技術性概述，我們參考綜述 (Yu et al., 2023) 及其中引用之文獻。

本節介紹的技術包括：Chain-of-Thought (CoT) Prompting、Chain-of-Thought Factored Decomposition Prompting、Tree-of-Thoughts (ToT) 及 Graph-of-Thoughts (GoT) Prompting，以及 Skeleton-of-Thought (SoT) Prompting。

Chain-of-Thought (CoT) Prompting

Chain-of-Thought (CoT) 提示促使模型以序列方式「思考」，透過逐步處理資訊 (Wei et al., 2022；Kojima et al., 2022；Lyu et al., 2023；Wang et al., 2023)。此技術不是直接跳到結論，而是提示模型暫停一下，將複雜的查詢拆解成可解釋的中間步驟。透過讓模型「展示其過程」並在每一步問題解決中進行邏輯推理，其解答的整體準確度通常會提升。

實務範例：你希望模型解一道數學題。與其簡單問模型「將 x^2 從 0 積分到 1 的結果是什麼？」不如在同一個提示中加入引導序列思考的指示，例如：「先定義函數。接著設定積分上下限。現在，逐步帶我走過積分過程。」這樣你可以追蹤模型的邏輯推進，且模型有時間適當處理每一步。

Chain-of-Thought Factored Decomposition Prompting

Chain-of-Thought Factored Decomposition Prompting 結合了 chain-of-thought 提示固有的序列推理與因子分解：將複雜的任務或概念拆解為較小且更易處理的部分 (Wang et al., 2022; Fu et al., 2022; Xue et al., 2023)。此技術透過引導模型進行序列化思考，同時分解並處理每個子組件，來精練模型的分析方法。

實務範例：你希望模型解釋光合作用的過程。與其直接提出問題（例如：「光合作用如何運作？」），你可以將提示結構化，先要求 LLM 定義光合作用所涉及的所有主要過程，然後再依序詳述各步驟。這有助於確保光合作用的每個面向都被充分拆解並以易懂的方式解釋。

Tree-of-Thoughts (ToT) 與 Graph-of-Thoughts (GoT) 提示

Tree-of-Thoughts (ToT) 和 Graph-of-Thoughts (GoT) 提示技術建立在 CoT 提示之上，超越線性推理，探索更多元且具創意的思考路徑。

ToT (Yao et al., 2023) 在 CoT 的基礎上建立一個決策樹，引導模型走過各種邏輯路徑並增強其決策過程。對於像頭腦風暴這類情境，ToT 使模型能夠模擬心智圖的過程，探索核心想法的多個分支。

類似地，GoT 將 LLM 提示視為圖形，便於進行複雜的思考聚合與操作 (Besta et al., 2023)。此方法允許更強的模組化與動態推理。

實務範例：你希望模型協助你頭腦風暴商業策略。在此情況下，可以從一個核心想法開始（例如改善客戶服務），然後要求 AI 產生多個分支策略或「如果...會怎樣？」的情境，以探索各種可能性。

Skeleton-of-Thought (SoT) Prompting

Skeleton-of-Thought (SoT) prompting 與人類解題方式類似，透過提供分段但完整的回應框架來引導回應 (Ning et al., 2023)。透過給模型一個結構化、高層次的「骨架」作為期望回應，模型能確保正確填充所提供的模板，而不會產生超出期望輸出範圍太多的內容。

實務範例：你想讓模型草擬一封商務電子郵件。既然你已經知道希望信件的結構，可以提供以下範本：「[Greeting], [Introduce topic], [Main content], [Closing remarks], [Signature].」如此一來，模型會被引導去填寫每一個所要求的段落，而不會遺漏範本中的重要部分或加入超出原先需求的內容。

2.2 情境理解與記憶

LLMs 已經發展出進階的情境理解與記憶能力，使它們能在長時間對話中維持相關性與連貫性。這項能力是透過各種創新技術實現，讓這些模型能回憶並參照先前互動，從而提供無縫的對話體驗。

本節介紹的技術包括：In-Context Prompting、Multi-Personas Prompting、Conversational Prompting，以及 Socratic Prompting。

In-Context Prompting

In-context prompting 透過在多次互動中維持上下文，模擬模型的一種「記憶」形式。此技術將歷史上下文整合到當前回應中，以利用注意力機制在增強 LLMs 序列一致性方面的重要性（Rubin et al., 2021；Dong et al., 2022；Min et al., 2021；Ye et al., 2022；Wang et al., 2022）。藉由在模型的當前上下文視窗內提供資訊，提示工程師能在先前引入的資訊基礎上構建更複雜的想法與情境，並產生更連貫且具上下文意識的回應。

實務範例：你正在與一個虛擬助理 LLM 討論行程規劃。你一開始提到你要前往巴黎，之後又詢問「推薦的餐廳」。由於對話發生在模型的上下文視窗內，模型會記住你先前的互動，並很可能建議巴黎的餐廳。

多重角色提示

多重角色提示利用 LLM 展現一致的角色或人格特質的能力，讓模型在互動中能「戴上不同的帽子」（Gu et al., 2023）。此技術可透過提供一個穩定的對話伙伴，同時又能適應不同語調、觀點與專業知識，來提升使用者體驗。

實務範例：你想看看莎士比亞與科學家在描述日落時會有多大的不同。你可能會先問：「莎士比亞會如何描述日落？」在收到回應後，你可能接著問：「那科學家會如何描述日落？」即便主題（日落）保持不變，模型仍能在莎士比亞的詩性人格與科學家的分析性人格之間切換，從而為你提供對同一主題的兩種不同感知。

對話式提示

會話式提示包含設計模擬自然人類對話的提示語（Wei et al., 2023; Liu et al., 2023）。這不僅提升模型回應的流暢度，也增加其相關性。當模型與使用者進行互動式對話時，例如詢問更多細節或建議後續問題，便能明顯看出這一點，從而豐富對話體驗。鼓勵來回的回應可以導致與模型之間更豐富、更具細微差異的交流，並標誌著人們將 AI 互動從單一查詢轉變為持續對話的轉變。

實際範例：你正在為即將到來的旅行打包。與其單問「雅典的天氣如何？」不如接著追問「我該為一整天觀光帶哪些衣物？」或「有沒有我應該注意的文化習俗？」以促成更完善、深入且自由的體驗。

蘇格拉底式提示

蘇格拉底式提示模擬蘇格拉底的對話法，透過提出一連串問題引導模型（或使用者）得出結論或領悟（Chang, 2023）。此技術讓使用者能透過探究特定感興趣的領域，來檢視一個 LLM 在某主題上知識的深度。

實務範例：你對正義的概念感到好奇。你可以先用一般性問題開始提示模型（例如：「什麼是正義？」），然後根據模型的回答再接著提出更細緻的問題，例如「它是否永遠與合法性一致？」。

2.3 具體性與目標設定

針對 LLMs 發展的具體性與目標設定技術，大幅提升了它們產出精確且以目標為導向回應的能力。透過使這些模型能區分不同類型的資訊傳達、聚焦於特定回應目標，並與整體目標對齊，此類方法在需要詳細且有目標資訊處理的多個領域中，極大地增進了 LLMs 的實用性與適用性。

本節介紹的技術包括：Show-me versus Tell-me Prompting、Target-your-response (TAR) Prompting、Prompt Macros 與 End-goal Planning，以及 Contrastive Prompting。

Show-me versus Tell-me Prompting

Show-me versus tell-me prompting 涉及指示 LLMs 要麼示範（“show”）要麼描述（“tell”）所指示的概念（Perez et al., 2021）。此方法的成敗取決於使用者辨別哪種類型的資訊對他們最有價值，或哪種輸出最適合手頭的任務。例如，「show-me」提示可能要求視覺示範，而「tell-me」提示則可能尋求對相同過程的描述性解釋。同時或接續使用兩種方法可以展示模型有效理解並適應對資訊傳達形式要求之微妙差異的能力。

實例說明：你想知道光合作用如何運作。你可以請模型「用圖示展示光合作用的運作方式」或「說明光合作用的過

程」。前者的請求可能會產生圖像化的表示，後者則較可能給出文字說明，而你對理想輸出形式的表述，以及對模型可能產生哪些輸出格式的預測，兩者都應該用來指導最初的請求。

將回應對準目標 (Target-your-response, TAR) 提示

將回應對準目標 (TAR) 提示指示模型將其回應聚焦於特定目標或目的，以提升輸出的相關性與簡潔性 (Zhao et al., 2021)。此方法在於細緻調整所期望答案的具體性。它強調明確指出你希望收到的回應格式或風格的重要性。

實務範例：你想進一步了解咖啡。與其僅僅問「告訴我關於咖啡的事」，你可以改成說：「提供一段 3 句話的摘要，說明咖啡的起源。」後者的提示更精準地引導模型至特定查詢，並為其回應提供更明確的上下文界限。

提示巨集與最終目標規劃

提示巨集與最終目標規劃利用預先定義的提示範本來建立模型需達成的整體目標。它們透過將可能的多個「微」提示與查詢結合成單一較大的「巨集」提示，確保 LLM 在每次互動中都朝向較廣的目標前進。在設計巨集提示時，必須確保提示夠廣泛以涵蓋所有預期微查詢的範圍，但又要足夠狹窄，使模型理解隱含請求的所有相關細節。

實務範例：您希望模型在一次互動中協助您規劃行程。與其逐一提出問題，您可以選擇使用像「規劃一個一月前往科羅拉多的五天行程」這類的巨集提示，模型較可能以適當的細緻度來理解此請求，並依序提供行程、住宿和活動建議。

對比提示

對比提示涉及要求模型比較或對照兩個概念、物件或想法。透過這種方式構造提示，模型的任務是辨識所提供主題之間的差異或相似處，從而對其特性與關係有更深入的理解。此方法利用模型辨別微妙差異的能力，使其能提供更具洞見且具情境相關性的回應。

實作範例：你想加深對兩種程式語言 Python 與 Java 差異的理解。你可以提示模型「比較並對照 Python 與 Java 程式語言」，以瞭解它們在語法、效能和使用情境上的差異。這種方法可以幫助你就特定專案做出明智的語言選擇，從而簡化開發流程並優化軟體的效能。

2.4 元認知與自我反思

元認知 (Wang and Zhao, 2023)、自我反思 (Shinn et al., 2023) 與自我解釋 (Gao et al., 2023) 領域探討 LLMs 的自我分析能力。透過賦予這些模型自我導向學習、預測使用者需求與生成程式碼的能力，這些方法不僅擴展了 LLMs 的應用範疇，還強化了它們的互動與協助功能，使其成為各種科技驅動領域中不可或缺的工具。

本節介紹的技術包括：自我反思提示 (Self-reflection Prompting)、元提示 (Meta-Prompting)、預期提示 (Anticipatory Prompting) 以及從提示到程式碼 (Prompt to Code)。

自我反思提示

自我反思提示允許模型評估自身的回應並批判性地處理其答案 (Shinn et al., 2023; Madaan et al., 2023; Xie et al., 2023)。當面對那些起初未以利於模型逐步推理方式提示的複雜任務時，這一方法特別有用。透過提示 LLM 對先前輸出進行自我反思，模型能檢閱並更新內容，使使用者得到更深思熟慮且更周延的回應。

實務範例：模型在一個倫理問題上給出令人存疑的答案。為了讓模型重新評估並為其初步答案補充更多細節，你可以接著問：「你確定嗎？」這會促使模型對先前回答進行自我反思，並可能導致對原始問題更深入的審視，進而產生更新的答案或對首次回答的詳細推理。

元提示

Meta-prompting 涉及引導 LLMs 反思其回應提示時的程序與方法論 (Wang and Zhao, 2023; Gao et al., 2023)。這種方法不僅強化模型在當前上下文視窗內對自身能力的理解，還透過鼓勵自我覺察與內省，提升其與 LLMs 互動的品質。

實務範例：你想知道如何撰寫更好的提示，因此你要求模型「分析過去給你的 5 個提示的有效性並提出改進建議」。既然提示的流程已被引入到當前上下文中，模型便可能提供可改進提示設計的見解，以從 LLMs 獲得更有利的輸出。

Anticipatory Prompting

Anticipatory prompting 使 AI 模型能根據當前上下文預見並回應未來的查詢或需求。此方法包含設計可促使模型同時提供直接答案與相關洞見的提示，從而預測可能的後續問題或疑慮。

實際範例：當被問到「我該如何種植番茄？」時，模型可能會從上下文推斷你是番茄種植新手，因此還會預先提供應對常見害蟲的建議，以回應可能的後續疑慮。

Prompt to Code

Prompt to code 著重在指示 AI 根據特定提示產生可運作的程式碼（Chen et al., 2023）。它利用模型訓練資料中所含的程式設計資訊，以便理解並依照使用者指定的語言、格式與其他要求產出程式碼片段。

實務範例：您希望模型為一個開發專案產生程式碼。您可以要求 LLM 「產生一個用於計算給定數字之費氏數列的 Python 函式。」在此指示下，模型即可依照您的規格產生相關程式碼。

2.5 方向性與回饋技術

方向性與回饋技術引導 LLM 朝特定任務前進或根據使用者回饋精練其回應。透過採用回應性回饋，使用者可以積極參與模型的學習過程，確保輸出更符合其期望與需求。與此同時，方向性刺激提示促進了引導輸出與 AI 自主性之間的平衡，鼓勵具有創意且量身訂做的回應。這些方法不僅精進使用者與模型之間的互動，亦大幅提升 LLM 在各種應用中的適應性與精確度。

本節介紹以下技術：回應式回饋提示、方向性刺激提示與模糊提示。

回饋導向提示

回饋導向提示將回饋直接納入提示流程，以提升 LLM 回應的品質與相關性（Zheng et al., 2023；Paul et al., 2023）。在此方法中，回饋會在模型的初次輸出之後立即給出，以引導後續的回應。

實務範例：你正在使用模型為一個標誌進行腦力激盪。模型產出的結果大致令人滿意，但你希望做出幾項修改。向模型提供「我喜歡這個配色，但設計能否更簡約一些？」這類回饋，會成為相關的輸入，讓模型在下一次輸出時更符合使用者的偏好。

方向性刺激提示

Directional stimulus prompting 涉及使用暗示或細微線索來引導 LLM 朝期望的方向前進，而不明確指示輸出內容（Shum et al., 2023）。當希望模型的回應具有意外性及更高創造力時，這項技術特別有用。透過僅提示使用者期望在輸出中出現的元素，模型便需自行填補空白並做出最佳猜測。

實例：你正在提示模型生成一個故事。與其詳細描述情節並得到完全符合你要求的故事，不如加入像「當英雄遇見公主時加入一個驚喜元素」這樣的提示會更有趣。這讓使用者表明他們希望發生某個出乎意料的事件，但不需要具體說明那個事件是什麼，並促使 LLM 在故事中融入未指明的轉折，同時讓使用者對細節保持懸念。

Ambiguous Prompting

模糊提示（Ambiguous prompting）是有意使用含糊不清的提示，以激發模型的創意或產生廣泛的回應。此方法類似於 Directional Stimulus Prompting，但故意設計得更加開放式，以鼓勵高度的創造力與非引導性的生成。當使用者不確定自己想從回應中得到什麼，或希望看到較為無偏見且未被過多提示影響的想法時，模糊提示最為適用。

實際範例：你正在提示模型生成一個故事。與其用具體的詳細提示要求特定敘事，例如「寫一個關於騎士從龍那裡救出公主的故事」，不如改為要求模型簡單地「寫一個關於勇氣的故事」。透過保持提示模糊，你減少了提供給 LLM 的上下文量，而是鼓勵生成不受特定使用者偏好影響的敘事。

2.6 多模態與跨領域技術

多模態與跨領域類別包含整合多種輸入模式與多樣知識領域的技術，以提升 LLMs 的多功能性與適用性。透過結合多種輸入類型並融合來自不同學科的知識，這些技術不僅培養 LLM 的深度與廣度

不僅改進回應品質，還為各種多元且複雜的情境開啟新應用的可能性，涵蓋從藝術創作到科學研究與歷史分析等領域。

本節介紹的技術包括：多模態提示、跨學科提示，以及歷史脈絡提示、視覺提示與模組化提示。

多模態提示

多模態提示（multimodal prompting）指的是在對 LLMs 進行提示時使用多種輸入類型（Bang et al., 2023；Lee et al., 2023）。現在能處理的不再只是文字，有些模型可以接收包含文字、影像、音訊檔案甚至影片等各種組合的提示，以協助為 LLM 的輸出提供上下文和引導。

實際範例：你希望模型針對一張特定的日落照片寫一首詩。你很在意模型能融入這張影像的具體細節。使用能接收多模態輸入的模型時，你可能會上傳那張日落照片，同時請求模型「透過寫一首詩來描述這個場景」。模型透過電腦視覺能力處理照片，並透過 LLM 處理使用者的請求，因此能理解視覺與文本輸入，以完成該請求。

跨領域提示（Cross-disciplinary Prompting）

跨領域提示（cross-disciplinary prompting）涉及將來自多個獨立學科的知識融合，以針對跨學科問題引導出獨特的解決方案（Cui et al., 2023；Harrison et al., 2023；Huang et al., 2022）。透過為 LLMs 裝備精心調整的工具與知識，它們在處理需要納入各領域見解的複雜查詢與任務時會更為熟練。這不僅提升了它們在這些特定領域的準確性與相關性，也進一步為它們在各種專業與學術領域的應用開闢了新可能。

實例：你是個熱衷小說的讀者，正在修讀物理課。你對某些較為複雜的科學主題感到吃力，因為這些概念在觀念上與你最感興趣的文學大相逕庭。你可能會提示模型「用古典文學的類比來解釋量子物理原理」。透過要求模型以另一個學科的視角來解釋某一學科，你期望模型對兩個領域都有足夠理解，能提供兼具細緻與準確的解釋，將不同概念串連起來，協助你釐清理解上的脈絡。

歷史情境、視覺與模組化提示

歷史情境、視覺與模組化提示技術著重將歷史背景、視覺元素與模組化結構融合進提示中（Liu et al., 2023）。歷史情境提示可能要求模型依特定歷史環境或參照來回答；視覺提示則可能使用影像來引導模型的回應；而模組化提示則利用結構化、以元件為基礎的提示來處理複雜的詢問。

實務範例：假設你想知道曾祖父如何看待現代科技，可以提示 LLM 「請以一位 1920 年代的人士能理解的方式描述國際網路。」藉由為輸出指定歷史情境（在此例為特定年代），模型能更好地為其回應建立脈絡。

2.7 創意與生成技術

創意與生成技術使 LLMs 能夠產生具創意的內容，並引導出多樣、創新且發人深省的回應。透過採用一系列提示策略，從鼓勵模糊性到施加詞彙限制，這些技術不僅擴展了 LLMs 在創意任務上的能力，也為其在敘事、教育內容生成與創意寫作等多元領域的應用開闢了新途徑。

本節介紹的技術包括：Flipped Interaction Prompting、Grammar Correction，以及 Constrained Vocabulary Prompting。

Flipped Interaction Prompting

翻轉互動提示顛覆了傳統的模型—使用者動態。模型不再由使用者主導並指示對話，而可能改為對使用者提出問題或尋求澄清。透過改變互動中預期的角色，模型能透過有針對性的提問引導使用者自行產出結果。這種來回互動有助於結構化的對話式腦力激盪會議，並允許使用者利用 LLM 促發自己回應的發展，從而對互動輸出保有完整的創意控制權。

實務範例：你正在擬定一份商業計劃。與其向模型提供細節讓其產生建議，使用者可以提示模型向他們逐一提問，以引導他們自行撰寫商業計劃，例如告訴模型「逐一問我問題以引導我製作一份完整的商業計劃。」作為回應，模型可能會開始詢問「你的商業主要目標是什麼？」或「你的目標客群是誰？」以促使使用者考慮商業計劃制定的各個面向。

文法更正

語法修正提示是給模型的具體指示，用以識別並修正使用者所提供文字中的語法錯誤或不一致之處。由於 LLMs 在大量語料上訓練，這類提示能夠利用模型精通的語言理解能力，作為會話式的語法檢查工具，根據所要求的輸出條件向使用者提供建議與改進。透過請求 LLM 改善措辭與寫作風格，使用者可以提升書寫溝通的品質與清晰度，並輕鬆將寫作調整為各種使用情境與語境。

實際範例：你正在撰寫一份正式報告，並希望確保語法與語氣適合該情境。你可能會請求模型「檢查下列報告的語法是否正確，並確保語言清晰且專業。」透過同時要求語氣與語法上的改進，模型便獲得適當的情境界限，得以分析並調整所提供的文字。

受限詞彙提示

受限詞彙提示（Constrained vocabulary prompting）是指將模型的回應限制在特定的詞彙集合或定義好的詞庫內。這種技術對模型的輸出施加具體的限制，確保所生成的內容遵循預先定義的詞彙表。透過限制詞彙，我們能更精確地控制語言生成過程，特別適用於需要精確性、遵守特定術語或避免敏感或不當用語的情境。

實際範例：假設你正在為醫療保健產業開發一個客服聊天機器人。為了確保機器人提供準確且一致的回應，你可以指示模型「僅使用醫學術語回答客戶問題，避免使用口語化語言」。此做法能促使模型維持專業且醫學上準確的語氣，降低在與尋求醫療相關資訊或協助的使用者互動時產生誤解或錯誤資訊的風險。

3 討論

在本工作中，我們將研究結果與現有文獻相互連結，以突顯實務提示工程的重要面向。儘管我們承認此領域的限制與其中大量仍在發展的議題，我們仍相信本分類框架能為實務者提供提示選擇的資源，並促進對於每種提示方法潛在使用情境的討論。透過檢視相關研究並考量可能的倫理挑戰與限制，本節整合了本調查的整體敘事，指引讀者朝向對 LLM 提示之細微面向的全面理解。

3.1 Related Work

先前已有多項調查將提示技術分類為各種框架。特別是軟體模式（Schmidt et al., 2013）已被用作解釋與詳述提示技術的框架，提供範例、核心概念、情境與使用動機（White et al., 2023）。由於這些軟體模式為重複出現的問題提供可重用的解決方案，提示技術通常非常適合這種公式化的拆解，特別是對於那些本質上結構化明顯或為較技術性使用情境設計的提示。

類似地，其他關於提示方法的目錄也對提示與基於提示的學習技術提供了技術性的檢視（Liu et al., 2023；Kanti Karmaker Santu and Feng, 2023），其中一些特別聚焦於依賴軟體系統之工程情境（Schmidt et al., 2023）。這些方法反映出在目前提示技術調查領域中，越來越多傾向於迎合具技術背景的從業者的廣泛趨勢。

由於像 ChatGPT 這類普遍可取得的對話介面，使 LLMs 廣受關注，良好提示設計的重要性在近年大幅增加（Van Dis et al., 2023；Bang et al., 2023；Yang et al., 2023）。數篇研究透過探討整體提示設計的新穎方法，為提示技術不斷演進的領域做出貢獻。例如，有些研究探討了

多個提示的彙總（Arora et al., 2022），而其他研究則指出零樣本提示方法顯著優於少樣本方法（Reynolds and McDonell, 2021），還有研究提出自動化提示生成作為傳統微調方法的一種無參數替代方案（Shin et al., 2020）。

此外，隨著進一步的研究致力於發掘預訓練 LLMs 的全部潛能，也促使人們檢視我們對 prompt 的當代理解以及由人主導的 prompt engineering 的角色。對於擴大版 LLMs 中 emergent abilities 的調查（Wei et al., 2022）、作為傳統微調方法之無參數替代的自動 prompt 生成（Shin et al., 2020），以及使用 Automatic Prompt Engineers（Zhou et al., 2022）等研究，合起來代表了對傳統 prompt 範式的重新評估。此外，像 chain-of-thought prompting 這類技術已被定期研究，作為在無需進一步使用者輸入的情況下提升 LLMs 在多步驟問題解決任務中推理能力的手段（Wei et al., 2022；Wang et al., 2022；Kojima et al., 2022；Wang et al., 2023）。這些研究共同涵蓋了多種策略，合力促成未來模型主導優化 LLMs 輸出以應用於廣泛場景的可能性。

3.2 挑戰與倫理考量

在提示設計中的倫理考量對大型語言模型的開發與應用至關重要（Tokayev, 2023）。雖然這些模型展現出令人印象深刻的的能力，但它們也帶來需要被妥善處理的重大倫理議題，以確保負責且安全的使用。其中一個核心問題涉及可能產生的偏見輸出（Bender et al., 2021；O'neil, 2017），反映了關於演算法公平性與倫理的討論。要解決此挑戰，不僅需謹慎檢視訓練資料與模型本身，還需細緻考量提示（prompts）的設計。

必須承認 LLMs 會受到其訓練資料的影響，這些資料可能會引入偏見。這些偏見可能以刻板印象、成見或對特定族群或觀念的不公平呈現形式出現 (Zhuo et al., 2023; Tokayev, 2023)。因此，提示設計者必須對潛在陷阱高度警覺，並主動致力於在提示與其生成的回應中減輕偏見。這包括仔細檢視與修訂提示，以避免強化有害的刻板印象或產生歧視性內容。

此外，道德提示 (ethical prompting) 不僅僅是避免偏見。它要求構思出與社會規範與價值觀相符的提示，促進 LLM 回應中的公平性與正面性 (Shaikh et al., 2022; Xu et al., 2023)。這種對齊旨在確保模型所產生的輸出符合倫理原則，並對個人與整體社會的福祉產生正面貢獻。設計者必須謹慎考量提示的後果，並努力引導 LLM 產出負責任且社會可接受的內容 (Ma et al., 2023)。

在提示設計領域，另一個倫理層面涉及處理非法與侵入性提示 (Deng et al., 2023)。非法提示指的是試圖獲取可能用於有害目的的資訊，例如與違法活動或會導致傷害的指示相關的查詢。侵入性提示則有可能侵犯隱私或保密性，這會帶來嚴重的倫理後果。提示設計者必須保持警覺，識別並防止此類提示，確保它們不會導致產生有害或不當的內容 (Wolf et al., 2023)。

為了強化倫理性提示 (prompting)，實施信任層 (trust layers) 至關重要。這些機制作為保障，確保 AI 生成的內容具備可靠性、可解釋性與可信度 (Zafar et al., 2023; Mökander et al., 2023)。信任層包括將 AI 回應與倫理標準、準確性基準及組織目標進行交叉檢核。它們提供額外的保證層，有助於維持對 AI 技術的信任，並促進在商業與技術等各領域中負責且有益的使用。

在為 LLMs 設計提示時的倫理考量，對其負責任且安全的部署至關重要。透過謹慎處理偏見、避免非法與侵入性提示，以及實施信任層，提示工程師可協助開發在道德責任與社會可接受範圍內運作的 AI 系統。這些措施不僅對維持對 AI 技術的信任是必要的，亦確保其對社會產生永續且正向的影響。

3.3 結語

在本次綜述中，我們梳理了對話式預訓練大型語言模型 (LLMs) 之提示技術與方法的全貌，並提供一份既適合聊天機器人類型模型的休閒使用者，也適合資深提示工程師的全面概覽。我們的目標是縮短學術界與實務應用之間的鴻溝，以非技術性的視角呈現研究與真實世界使用中普遍存在的多樣方法。不同於可能深入探討技術細節的傳統綜述，我們的分類設計旨在與更廣泛且跨領域的讀者產生共鳴，確保即使沒有人工智慧與機器學習專業背景的人，也能理解此處呈現的基本概念與技術。

我們的調查透過提供一種對提示技巧進行分類的新穎觀點，對大型語言模型領域做出有價值的貢獻。雖然我們在可及性與跨領域吸引力方面保持承諾，但我們的分類重要性不僅止於其使用者友好的方法。透過提供一個理解並運用這些技巧的結構化框架，我們旨在使更廣泛的使用者能夠在各自領域中發揮 LLM 的變革性潛力。此外，我們的分類還鼓勵更深入地探索當多元觀點與背景與這些模型的能力交互時，可能產生的創意應用與創新解決方案。本質上，本工作作為促進跨領域合作與推動語言模型使用民主化的催化劑，最終豐富人類與 AI 互動的整體格局。

參考文獻

- [1] M. Abbasian, I. Azimi, A. M. Rahmani, and R. Jain. Conversational health agents: A personalized LLM-powered agent framework. arXiv preprint arXiv:2310.02374, 2023.
- [2] S. Arora、A. Narayan、M. F. Chen、L. Orr、N. Guha、K. Bhatia、I. Chami、F. Sala、及 C. Ré。Ask me anything：一種針對語言模型的簡單提示策略。arXiv preprint arXiv:2210.02441，2022。
- [3] Y. Bang、S. Cahyawijaya、N. Lee、W. Dai、D. Su、B. Wilie、H. Lovenia、Z. Ji、T. Yu、W. Chung 等人。對 ChatGPT 在推理、產生幻覺與互動性方面之多任務、多語言、多模態評估。arXiv preprint arXiv:2302.04023，2023。
- [4] E. M. Bender、T. Gebru、A. McMillan-Major、及 S. Shmitchell。關於「隨機鸚鵡」的危險：語言模型會不會太大？收錄於 2021 年 ACM 公平性、問責性與透明度會議論文集，頁 610–623，2021。
- [5] M. Besta、N. Blach、A. Kubicek、R. Gerstenberger、L. Gianinazzi、J. Gajda、T. Lehmann、M. Podstawski、H. Niewiadomski、P. Nyczyk 等人。Graph of thoughts：以大型語言模型解決複雜問題。arXiv preprint arXiv:2308.09687，2023。
- [6] R. Bommasani, D. A. Hudson, E. Adeli, R. Altman, S. Arora, S. von Arx, M. S. Bernstein, J. Bohg, A. Bosselut, E. Brunskill, et al. 關於基礎模型的機會與風險。arXiv preprint arXiv:2108.07258，2021。
- [7] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. 語言模型是少量示例學習者。Advances in Neural Information Processing Systems, 33: 1877-1901，2020。
- [8] E. Y. Chang. 使用蘇格拉底式方法提示大型語言模型。收錄於 2023 IEEE 第十三屆年度計算與通訊研討會與會議

(CCWC), 頁 0351-0360。IEEE, 2023。

[9] X. Chen, M. Lin, N. Schärli, and D. Zhou. 教導大型語言模型自我除錯。arXiv preprint arXiv:2304.05128, 2023。

[10] A. Chowdhery、S. Narang、J. Devlin、M. Bosma、G. Mishra、A. Roberts、P. Barham、H. W. Chung、C. Sutton、S. Gehrmann 等人。PaLM：以 pathways 擴展語言模型規模。Journal of Machine Learning Research, 24 (240):1-113, 2023。

[11] J. Cui、Z. Li、Y. Yan、B. Chen、L. Yuan。Chatlaw：整合外部知識庫的開源法律大型語言模型。arXiv preprint arXiv:2306.16092, 2023。

[12] Y. Deng、W. Zhang、S. J. Pan、L. Bing。大型語言模型中的多語言越獄挑戰。arXiv preprint arXiv:2310.06474, 2023。

[13] J. Devlin、M.-W. Chang、K. Lee、K. Toutanova。BERT：用於語言理解的深度雙向 Transformer 預訓練。arXiv preprint arXiv:1810.04805, 2018。

[14] G. Dong、J. Zhao、T. Hui、D. Guo、W. Wang、B. Feng、Y. Qiu、Z. Gongque、K. He、Z. Wang 等。重新檢視 LLMs 的輸入擾動問題：用於嘈雜槽位填充任務的統一魯棒性評估框架。收錄於 CCF 國際自然語言處理與中文計算會議，頁 682–694。Springer, 2023。

[15] Q. Dong、L. Li、D. Dai、C. Zheng、Z. Wu、B. Chang、X. Sun、J. Xu、Z. Sui。關於 in-context learning 的綜述。arXiv preprint arXiv:2301.00234, 2022。

[16] Y. Fu、H. Peng、A. Sabharwal、P. Clark、T. Khot。基於複雜度的提示法，用於多步推理。arXiv preprint arXiv:2210.00720, 2022。

[17] H. Gao、T.-E. Lin、H. Li、M. Yang、Y. Wu、W. Ma、Y. Li。自我解釋提示改善大型語言模型的對話理解。arXiv preprint arXiv:2309.12940, 2023。

[18] H. Gu、C. Degachi、U. Genç、S. Chandrasegaran、以及 H. Verma。透過提示 (prompting) 建立對話代理人格的有效性。arXiv preprint arXiv:2310.11182, 2023。

[19] R. M. Harrison、A. Dereventsov、以及 A. Bibin。使用預訓練大型語言模型進行多模態引導的零樣本推薦。arXiv preprint arXiv:2309.01026, 2023。

[20] W. Huang、F. Xia、T. Xiao、H. Chan、J. Liang、P. Florence、A. Zeng、J. Tompson、I. Mordatch、Y. Chebotar 等。內在獨白：透過與語言模型的規劃進行具身推理 (Embodied reasoning)。arXiv preprint arXiv:2207.05608, 2022。

[21] A. Q. Jiang、A. Sablayrolles、A. Mensch、C. Bamford、D. S. Chaplot、D. d. l. Casas、F. Bressand、G. Lengyel、G. Lample、L. Saulnier 等。Mistral 7B。arXiv preprint arXiv:2310.06825, 2023。

[22] S. Kanti Karmaker Santu 和 D. Feng。TELeR：一個用於基準測試複雜任務的 LLM 提示通用分類法。arXiv preprint arXiv:2305.11430, 2023。

[23] E. Kasneci、K. Seßler、S. Küchemann、M. Bannert、D. Dementieva、F. Fischer、U. Gasser、G. Groh、S. Günnemann、E. Hüllermeier 等。ChatGPT 對公益有幫助？關於大型語言模型在教育領域的機會與挑戰。Learning and Individual Differences, 103:102274, 2023。

[24] T. Kojima、S. S. Gu、M. Reid、Y. Matsuo 和 Y. Iwasawa。大型語言模型是零樣本推理者。Advances in Neural Information Processing Systems, 35:22199-22213, 2022。

[25] T. H. Kung、M. Cheatham、A. Medenilla、C. Sillos、L. De Leon、C. Elepaño、M. Madriaga、R. Aggabao、G. DiazCandido、J. Maningo 等。ChatGPT 在 USMLE 上的表現：使用大型語言模型進行 AI 輔助醫學教育的潛力。PLoS Digital Health, 2(2):e0000198, 2023。

[26] Y.-L. Lee、Y.-H. Tsai、W.-C. Chiu 與 C.-Y. Lee。以缺失模態進行視覺辨識的多模態提示。發表於 IEEE/CVF Computer Vision and Pattern Recognition 會議論文集，頁 14943-14952, 2023 年。

[27] P. Liu、W. Yuan、J. Fu、Z. Jiang、H. Hayashi 與 G. Neubig。預訓練、提示與預測：自然語言處理中提示方法的系統性綜述。ACM Computing Surveys, 55(9):1-35, 2023 年。

[28] S. Liu、C. Peng、C. Wang、X. Chen 與 S. Song。icsberts：在智慧客服中優化預訓練語言模型。Procedia Computer Science, 222:127-136, 2023 年。

[29] X.-Y. Liu、G. Wang 與 D. Zha。FinGPT：為金融大型語言模型普及網路規模資料。arXiv preprint arXiv:2307.10485, 2023 年。

[30] Q. Lyu, S. Havaldar, A. Stein, L. Zhang, D. Rao, E. Wong, M. Apidianaki, and C. Callison-Burch。Faithful chain-of-thought reasoning。arXiv preprint arXiv:2301.13379, 2023。

[31] P. Ma, Z. Li, A. Sun, and S. Wang。"oops, did i just say that?" testing and repairing unethical suggestions of large language models with suggest-critique-reflect process。arXiv preprint arXiv:2305.02626, 2023。

[32] A. Madaan, N. Tandon, P. Gupta, S. Hallinan, L. Gao, S. Wiegrefe, U. Alon, N. Dziri, S. Prabhume, Y. Yang, et al。Self-refine: Iterative refinement with self-feedback。arXiv preprint arXiv:2303.17651, 2023。

- [33] S. Min, M. Lewis, L. Zettlemoyer, and H. Hajishirzi. Metaicl: Learning to learn in context. arXiv preprint arXiv:2110.15943, 2021.
- [34] J. Mökander, J. Schuett, H. R. Kirk 與 L. Floridi. 《Auditing large language models: a three-layered approach》. *AI and Ethics*, 頁 1–31, 2023.
- [35] X. Ning, Z. Lin, Z. Zhou, H. Yang 與 Y. Wang. 《Skeleton-of-thought: Large language models can do parallel decoding》. arXiv preprint arXiv:2307.15337, 2023.
- [36] C. O’neil. 《Weapons of math destruction: How big data increases inequality and threatens democracy》. Crown, 2017.
- [37] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray 等人. 《Training language models to follow instructions with human feedback》. *Advances in Neural Information Processing Systems*, 35: 27730–27744, 2022.
- [38] D. Paul, M. Ismayilzada, M. Peyrard, B. Borges, A. Bosselut, R. West, and B. Faltings. REFINER: Reasoning feedback on intermediate representations. arXiv preprint arXiv:2304.01904, 2023.
- [39] E. Perez, D. Kiela, and K. Cho. True few-shot learning with language models. *Advances in neural information processing systems*, 34:11054–11070, 2021.
- [40] F. Petroni, T. Rocktäschel, P. Lewis, A. Bakhtin, Y. Wu, A. H. Miller, and S. Riedel. Language models as knowledge bases? arXiv preprint arXiv:1909.01066, 2019.
- [41] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.
- [42] L. Reynolds 與 K. McDonell. Prompt programming for large language models: Beyond the few-shot paradigm. 刊於 2021 CHI Conference on Human Factors in Computing Systems 擴展摘要, 頁 1–7, 2021.
- [43] B. Roziere, J. Gehring, F. Gloeckle, S. Sootla, I. Gat, X. E. Tan, Y. Adi, J. Liu, T. Remez, J. Rapin 等人. Code llama: Open foundation models for code. arXiv preprint arXiv:2308.12950, 2023.
- [44] O. Rubin, J. Herzig 與 J. Berant. Learning to retrieve prompts for in-context learning. arXiv preprint arXiv:2112.08633, 2021.
- [45] T. Schick 與 H. Schütze. It’s not just size that matters: Small language models are also few-shot learners, 2021.
- [46] D. C. Schmidt, M. Stal, H. Rohnert, F. Buschmann. 《Pattern-oriented software architecture, patterns for concurrent and networked objects, volume 2》. John Wiley & Sons, 2013.
- [47] D. C. Schmidt, J. Spencer-Smith, Q. Fu, J. White. 朝向提示範式目錄以強化提示工程的規範. <https://www.dre.vanderbilt.edu/~schmidt/PDF/ADA-User-Journal.pdf>, 2023.
- [48] M. Sclar, Y. Choi, Y. Tsvetkov, A. Suhr. 量化語言模型對提示設計中虛假特徵的敏感度, 或: 我如何開始擔心提示格式化. arXiv preprint arXiv:2310.11324, 2023.
- [49] O. Shaikh, H. Zhang, W. Held, M. Bernstein, D. Yang. 再想一想, 我們還是別逐步思考了! 零樣本推理中的偏見與毒性. arXiv preprint arXiv:2212.08061, 2022.
- [50] T. Shin, Y. Razeghi, R. L. Logan IV, E. Wallace 與 S. Singh. AutoPrompt: 以自動生成的提示詞從語言模型引出知識. arXiv preprint arXiv:2010.15980, 2020.
- [51] N. Shinn, B. Labash 與 A. Gopinath. Reflexion: 具有動態記憶與自我反省的自主代理. arXiv preprint arXiv:2303.11366, 2023.
- [52] K. Shum, S. Diao 與 T. Zhang. 結合標註資料的 chain-of-thought 進行自動提示增強與選擇. arXiv preprint arXiv:2302.12822, 2023.
- [53] A. J. Thirunavukarasu, D. S. J. Ting, K. Elangovan, L. Gutierrez, T. F. Tan 與 D. S. W. Ting. 醫學領域中的大型語言模型. *Nature Medicine*, 29(8):1930–1940, 2023.
- [54] R. Thoppilan, D. De Freitas, J. Hall, N. Shazeer, A. Kulshreshtha, H. Cheng, A. Jin, T. Bos, L. Baker, Y. Du, et al. LaMDA: Language models for dialog applications. arXiv preprint arXiv:2201.08239, 2022.
- [55] K.-J. Tokarev. Ethical implications of large language models a multidimensional exploration of societal, economic, and technical concerns. *International Journal of Social Analytics*, 8(9):17–33, 2023.
- [56] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. LLaMA: Open and efficient foundation language models. arXiv preprint arXiv:2302.13971, 2023.
- [57] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, N. Bashlykov, S. Batra, P. Bhargava, S. Bhosale, et al. LLaMA 2: Open foundation and fine-tuned chat models. arXiv preprint arXiv:2307.09288, 2023.
- [58] E. A. Van Dis, J. Bollen, W. Zuidema, R. van Rooij, C. L. Bockting. Chatgpt: 研究的五項優先事項. *Nature*, 614(7947): 224–226, 2023.
- [59] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.
- [60] B. Wang, X. Deng, H. Sun. Iteratively prompt pre-trained language models for chain of thought. arXiv

preprint arXiv:2203.08383, 2022。

[61] J. Wang、Q. Sun、N. Chen、X. Li、M. Gao。Boosting language models reasoning with chain-of-knowledge prompting。arXiv preprint arXiv:2306.06427, 2023。

[62] K. Wang、F. Duan、S. Wang、P. Li、Y. Xian、C. Yin、W. Rong 及 Z. Xiong。《Knowledge-driven CoT: Exploring faithful reasoning in LLMs for knowledge-intensive question answering》。arXiv preprint arXiv:2308.13259, 2023。

[63] X. Wang、J. Wei、D. Schuurmans、Q. Le、E. Chi、S. Narang、A. Chowdhery 及 D. Zhou。《Self-consistency improves chain of thought reasoning in language models》。arXiv preprint arXiv:2203.11171, 2022。

[64] Y. Wang 及 Y. Zhao。《Metacognitive prompting improves understanding in large language models》。arXiv preprint arXiv:2308.05342, 2023。

[65] J. Wei、Y. Tay、R. Bommasani、C. Raffel、B. Zoph、S. Borgeaud、D. Yogatama、M. Bosma、D. Zhou、D. Metzler 等人。《Emergent abilities of large language models》。arXiv preprint arXiv:2206.07682, 2022。

[66] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in Neural Information Processing Systems*, 35:24824-24837, 2022.

[67] J. Wei, S. Kim, H. Jung, and Y.-H. Kim. Leveraging large language models to power chatbots for collecting user self-reported data. arXiv preprint arXiv:2301.05843, 2023.

[68] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt. A prompt pattern catalog to enhance prompt engineering with ChatGPT. arXiv preprint arXiv:2302.11382, 2023.

[69] Y. Wolf, N. Wies, Y. Levine, and A. Shashua. Fundamental limitations of alignment in large language models. arXiv preprint arXiv:2304.11082, 2023.

[70] S. Wu、O. Irsoy、S. Lu、V. Dabravolski、M. Dredze、S. Gehrmann、P. Kambadur、D. Rosenberg 與 G. Mann。BloombergGPT：一個用於金融的大型語言模型。arXiv preprint arXiv:2303.17564, 2023。

[71] Y. Xie、K. Kawaguchi、Y. Zhao、X. Zhao、M.-Y. Kan、J. He 與 Q. Xie。分解透過自我評估導引解碼提升推理。arXiv preprint arXiv:2305.00633, 2023。

[72] F. F. Xu、U. Alon、G. Neubig 與 V. J. Hellendoorn。大型程式碼語言模型的系統性評估。發表於第 6 屆 ACM SIGPLAN 機器程式設計國際研討會論文集，頁 1–10, 2022。

[73] X. Xu、K. Kong、N. Liu、L. Cui、D. Wang、J. Zhang 與 M. Kankanhalli。一個 LLM 可能自我欺騙：基於提示的對抗性攻擊。arXiv preprint arXiv:2310.13345, 2023。

[74] T. Xue、Z. Wang、Z. Wang、C. Han、P. Yu 與 H. Ji。RCOT：透過反轉思路鏈（reversing chain-of-thought）來偵測並修正推理中的事實不一致。arXiv preprint arXiv:2305.11499, 2023。

[75] J. Yang、H. Jin、R. Tang、X. Han、Q. Feng、H. Jiang、B. Yin 與 X. Hu。在實務中善用 LLMs 的力量：關於 ChatGPT 及其延伸的綜述。arXiv preprint arXiv:2304.13712, 2023。

[76] L. Yang、H. Chen、Z. Li、X. Ding 與 X. Wu。ChatGPT 不足以應對一切：以知識圖譜強化大型語言模型以進行具事實意識的語言建模。arXiv preprint arXiv:2306.11489, 2023。

[77] S. Yao、D. Yu、J. Zhao、I. Shafran、T. L. Griffiths、Y. Cao 與 K. Narasimhan。思緒樹（Tree of thoughts）：使用大型語言模型進行有目的的問題解決。arXiv preprint arXiv:2305.10601, 2023。

[78] X. Ye, S. Iyer, A. Celikyilmaz, V. Stoyanov, G. Durrett, 和 R. Pasunuru。Complementary explanations for effective in-context learning。arXiv preprint arXiv:2211.13892, 2022。

[79] Z. Yu, L. He, Z. Wu, X. Dai, 和 J. Chen。Towards better chain-of-thought prompting strategies: A survey。arXiv preprint arXiv:2310.04959, 2023。

[80] A. Zafar, V. B. Parthasarathy, C. L. Van, S. Shahid, A. Shahid, 等。Building trust in conversational ai: A comprehensive review and solution architecture for explainable, privacy-aware systems using LLMs and knowledge graph。arXiv preprint arXiv:2308.13534, 2023。

[81] J. Zamfirescu-Pereira, R. Y. Wong, B. Hartmann, 和 Q. Yang。Why Johnny can't prompt: how non-AI experts try (and fail) to design LLM prompts。In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, pages 1-21, 2023。

[82] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, et al. 大型語言模型概論調查。arXiv preprint arXiv:2303.18223, 2023.

[83] Z. Zhao, E. Wallace, S. Feng, D. Klein, and S. Singh. 使用前先校準：提升語言模型少量示例表現。收錄於 *International Conference on Machine Learning*, 頁 12697-12706。PMLR, 2021.

[84] C. Zheng, Z. Liu, E. Xie, Z. Li, and Y. Li. 逐步提示（Progressive-hint prompting）能改善大型語言模型的推理。arXiv preprint arXiv:2304.09797, 2023.

[85] C. Zhou, Q. Li, C. Li, J. Yu, Y. Liu, G. Wang, K. Zhang, C. Ji, Q. Yan, L. He, et al. 關於預訓練基礎模型的綜合調查：從 BERT 到 ChatGPT 的發展史。arXiv preprint arXiv:2302.09419, 2023.

[86] Y. Zhou, A. I. Muresanu, Z. Han, K. Paster, S. Pitis, H. Chan, and J. Ba. Large language models are human-level prompt engineers. arXiv preprint arXiv:2211.01910, 2022.

[87] T. Y. Zhuo, Y. Huang, C. Chen, and Z. Xing. Exploring ai ethics of chatgpt: A diagnostic analysis. arXiv preprint arXiv:2301.12867, 2023.

Preprint。此研究於第 4 屆國際人工智慧、機器學習、資料科學與機器人學會議（2023）發表。

¹ <https://platform.openai.com/docs/guides/prompt-engineering>