

聊天機器人學習鷹架對程式設計教材與程式碼理解之比較

Comparison of programming textbook and code comprehension based on chatbot

scaffolding

楊沛庭，郭瀚中，張智凱
臺南大學數位學習科技學系
chihkai@mail.nutn.edu.tw

【摘要】本研究探討如何提升初學者學習程式設計的成效，比較運用聊天機器人為教科書學習鷹架與程式理解練習兩種教學策略。結果顯示，教科書學習鷹架能透過逐步引導，降低學生閱讀程式設計教科書時的認知負荷，幫助他們掌握程式設計的基礎概念。而程式理解練習則著重於加深學生對程式邏輯與結構的掌握，更適合進階學習者。此外，研究發現聊天機器人能提供即時且客製化的學習支持，不僅緩解了學生的學習焦慮，還有效提升了學習信心與動機。綜合研究結果顯示，結合傳統教學策略與AI學習科技，能創造更高效的學習環境，進一步增強學生的程式設計能力。

【關鍵詞】學習鷹架；程式理解；認知負荷；聊天機器人；程式設計教育

Abstract: This study explored improving programming learning outcomes for beginners by comparing two teaching strategies: using a chatbot as a textbook learning scaffold and practicing programming comprehension. The results showed that chatbot-based scaffolds reduce cognitive load through step-by-step guidance, helping students master basic programming concepts. Program comprehension exercises focus on deepening understanding of logic and structure, making them more suitable for advanced learners. Additionally, chatbots provide instant, customized support, alleviating learning anxiety and enhancing confidence and motivation. Overall, the study suggests that integrating AI learning tools with traditional teaching strategies can create a more efficient learning environment and further enhance students' programming capabilities.

Keywords: Learning Scaffolding, Program Comprehension, Cognitive Load, Chatbot, Programming Education

1. 前言

程式設計是一個充滿挑戰的領域，初學者需掌握語法、結構，並將問題轉化為程式碼（Winkler & Flatscher, 2023）。這過程不僅要求邏輯思維與抽象概念，還需大量時間與練習。然而，學生常因缺乏指導而感到畏懼，進而影響其資訊科技領域的發展。因此，教育者需尋求有效的教學方法，以增進學生的學習動機與能力（許育嘉，2021）。

本研究目的為探討結合學習鷹架理論、程式理解練習和聊天機器人輔助，以提升初學者學習程式設計的成效。學習鷹架的應用被認為是一種有效的輔助工具（Peng et al., 2022），能幫助學生在面對複雜任務時聚焦於關鍵概念與技能，並透過結構化支持，逐步掌握新知識。聊天機器人能根據學生的進度與需求調整教學內容（Alwazzan, 2024），提供針對性的支援，不僅緩解了學生的學習焦慮，還有效提升了學習信心與動機。因此，本研究比較運用聊天機器人為教科書學習鷹架（以下簡稱**機器人教材鷹架**）與程式理解練習對降低認知負荷與提升學習成效的效果，並將採用認知負荷測量問卷及 Orange Data Mining 做為研究工具，探討不同教學策略如何影響學生的認知負荷，以及聊天機器人輔助在優化學習過程中的作用。

2. 文獻探討

2.1. 程式講解練習與學習鷹架

程式設計的抽象性與複雜性使初學者在理解語法、邏輯結構及除錯時感到困惑，學習動機不足可能來自興趣不高或缺乏實作機會（張基成、林冠佑，2016）。為提升學習成效，程式講解練習與學習鷹架的結合可有效降低學生負擔並促進理解。程式講解練習透過 AI 輔助學習，能自動分析程式碼，提供逐行說明，幫助學生掌握邏輯與語法，減少挫折並提升動機。這些工具利用自然語言處理技術，將程式碼轉換為易懂的描述，使非程式設計背景的學生也能理解其功能。此外，學習鷹架源自 Lev Vygotsky 的「近側發展區」理論（ZPD），強調適時支持以促進學生的獨立學習（Rahman, 2024）。程式設計教師可透過視覺化工具、實作、與小組討論，逐步引導學生掌握知識並撤去支持，強化運算思維與程式理解能力。研究顯示，後設認知鷹架與支持性鷹架特別能幫助先備知識較低的學生（陳明溥等，2022）。整合 AI 程式講解與學習鷹架，不僅能提升學習效率與理解能力，還能增強學生的解決問題能力與自信心，為程式設計教學提供更有效的策略。

2.2. 認知負荷

認知負荷理論（Cognitive Load Theory, CLT）用以解釋學習中認知資源的使用及其對學習成效的影響。該理論指出，當學習所需的認知資源超過工作記憶容量時，會導致認知負荷過重，影響學習效果（Sweller, 1988）。認知負荷分為內在負荷、外在負荷和增生負荷：內在負荷與內容複雜性及學生先備知識相關；外在負荷來自不良教學設計；增生負荷則是促進長期記憶的心智努力。認知負荷理論不僅為教育工作者提供了設計有效教學活動的依據，也強調了學生先備知識的重要性，鼓勵教師根據學生的特點調整教學內容和方式，以達到最佳的學習效果。

3. 研究方法

3.1. 研究架構—學習活動設計

本研究將在 JAVA 程式設計與實作課程中實施 16 週的課程活動，對象為 24 名大三學生，其中女性有 6 人，教材採用由 Y. Daniel Liang 所著的《Introduction to Java Programming Comprehensive Version 10th Edition》。課程期間以抽籤隨機分組，而教師依據課程進度，隨機分配程式碼或教科書內容的任務。學生可使用聊天機器人（如 DeepSeek、ChatGPT）輔助理解，聊天機器人的回饋與互動模式主要為文字對話，其功能包含程式碼或教科書段落解釋、概念釐清及鷹架輔助，隨後學生需上台講解並回答提問。換言之，即是比較學習者的**講解教材任務與講解程式碼任務**的認知負荷。課程結束後，學生填寫認知負荷問卷，針對兩種講解內容進行評估，並透過結果分析學生在聊天機器人輔助學習下的認知負荷差異。本研究不僅能了解不同學習內容對學生認知負荷的影響，也期望能為程式設計課程提供新的教學策略，提升學習效率與成效。

3.2. 研究工具

本研究將採用 Leppink 等人（2013）開發的認知負荷測量問卷，以評估學生在學習任務中經歷的不同類型認知負荷。該問卷基於認知負荷理論具有良好的信效度。問卷收集完成後將使用 Orange Data Mining 作為問卷分析工具，這是一款功能強大的開源資料分析及機器學習工具，廣泛應用於研究和教育中（Demšar et al., 2013）。透過這兩款工具，本研究將分析學生在機器人教材鷹架與程式碼理解過程中的認知負荷差異。此外，問卷最後將加入學生自我能力評估的開放性問題，以進一步蒐集學習經驗反饋，從而優化教學策略並提升學習成效。

3.3. 研究假設

基於上述文獻探討與研究目標，本研究針對機器人教材鷹架與程式理解練習在程式設計教育中的應用進行分析，提出以下假設，以探討不同教學策略對學生認知負荷的影響，並進一步檢視聊天機器人輔助在優化學習過程中的作用：

1. 程式理解練習的認知負荷高於機器人教材鷹架，因其操作過程需集中處理邏輯與語法，對學生的工作記憶需求更高。
2. 聊天機器人的輔助功能可降低學生在程式理解練習中的認知負荷並提升學習信心。

4. 研究結果與分析

● 不同學習內容對認知負荷的影響，程式碼與教科書講解的比較

根據圖 1 所示，講解教科書的心力耗費 ($M = 3.29$, $SD = 0.31$) 與困難度感知 ($M = 2.88$, $SD = 0.25$) 略高於講解程式碼 ($M = 3.17$, $SD = 0.31$; $M = 2.79$, $SD = 0.27$)，這可能因為教科書內容較具理論性，對抽象理解能力要求較高。然而，講解教科書的挫折感 ($M = 2.25$, $SD = 0.29$) 低於講解程式碼 ($M = 2.42$, $SD = 0.31$)，顯示教科書的結構化與清晰編排有助於減少學習壓力，對初學者特別有利。在專注力需求方面，講解程式碼 ($M = 3.42$, $SD = 0.25$) 略高於講解教科書 ($M = 3.38$, $SD = 0.27$)，反映程式碼學習需要更高的注意力，可能因其高度操作性與即時回饋特性，使學生需集中精力處理邏輯推理與細節。然而，這種高專注力需求可能對基礎薄弱者造成學習壓力，因此建議提供分步指導或範例分析以協助適應。本研究發現，在聊天機器人輔助下，講解教科書的認知負荷反而高於講解程式碼，此結果與預期不符，可能原因為教科書內容較為抽象，學生需投入更多認知資源進行組織與理解，而程式碼學習則因具體操作性較強，使學生能更快速建立概念。為優化學習體驗，建議教科書設計納入情境範例與互動模擬工具，以強化理論與實踐的連結，提高學習成效。



圖 1、認知負荷問卷之平均值統計

● 聊天機器人輔助對學習信心的提升有顯著效果

在聊天機器人輔助下，學生對程式設計能力的自我評估顯著提高。問卷中自我能力評估的開放性問題結果顯示，學生認為聊天機器人在解決疑難問題和提供即時反饋方面發揮關鍵作用，增強了學習信心。具體而言，學生在聊天機器人輔助之前的自我評分平均為 2.54，而在有聊天機器人輔助後平均評分為 3.67。這表明人工智慧輔助工具能緩解學習焦慮，特別是在複雜任務中。建議未來優化聊天機器人的推理和解釋能力，並結合多媒體展示，以支持個性化學習需求。

5. 結論

本研究探討機器人教材鷹架與程式理解練習在降低認知負荷及提升學習成效上的效果，並結合聊天機器人的輔助功能進一步分析其對學生學習過程的影響。研究結果顯示，講解教科書相較於講解程式碼，雖在心力耗費上略高，但因結構化內容與理論性編排，使學生在情緒負荷上表現更低，特別是在挫折感的降低上更具優勢。另一方面，講解程式碼因操作性強及即時反饋特性，對專注力的需求明顯較高，適合進階學習者進行深入理解與挑戰性任務的學習。此外，兩種教學方式在提升學生學習成效上均有效果，學生的程式設計能力自我評估從學習前的低水準有所提升，展現出機器人教材鷹架與程式理解練習在促進知識吸收與應用

上的重要性。同時，聊天機器人作為輔助工具，透過即時解答與互動性指導，減少學生的學習焦慮並增強信心，使其在面對複雜問題時具備更高的自我效能感。綜上所述，本研究不僅驗證了機器人教材鷹架與程式理解練習的有效性，也強調了 AI 在優化教學過程與提升學習效率中的潛在價值，為未來程式設計教育的教學策略提供了具體參考。

6. 致謝

感謝教育部教學實踐研究計畫經費贊助，計畫編號 PSK1133865。

參考文獻

- 張基成、林冠佑（2016）。從傳統數位學習到遊戲式數位學習－學習成效、心流體驗與認知負荷。科學教育學刊，24(3)，221-248。 <https://doi.org/10.6173/CJSE.2016.2403.01>
- 陳明溥、王麗君、豐佳燕（2022）。先備知識與鷹架策略對程式設計初學者運算思維概念、技能實踐與態度之影響。數位學習科技期刊，14(4)，137-166。
<https://doi.org/10.53106/2071260X2022101404006>
- 許育嘉（2021）。高等教育程式設計課程教學之多元化發展與變革。臺灣教育評論月刊，10(1)，68-75。
- Alwazzan, M. S. (2024). Investigating the effectiveness of artificial intelligence chatbots in enhancing digital dialogue skills for students. *European Journal of Educational Research*, 13(2), 573-584. <https://doi.org/10.12973/eu-jer.13.2.573>
- Demšar, J., Curk, T., Erjavec, A., Gorup, Č., Hočevár, T., Milutinovič, M., ... & Zupan, B. (2013). Orange: data mining toolbox in Python. *the Journal of machine Learning research*, 14(1), 2349-2353.
- Peng, J., Yuan, B., Sun, M., Jiang, M., & Wang, M. (2022). Computer-Based Scaffolding for Sustainable Project-Based Learning: Impact on High- and Low-Achieving Students. *Sustainability*, 14(19), 12907. <https://doi.org/10.3390/su141912907>
- Rahman, L. (2024). Vygotsky's zone of proximal development of teaching and learning in STEM education. *International Journal of Engineering Research & Technology*, 13(8).
- Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive science*, 12(2), 257-285. [https://doi.org/10.1016/0364-0213\(88\)90023-7](https://doi.org/10.1016/0364-0213(88)90023-7)
- Winkler, T., & Flatscher, R. G. (2023). Cognitive Load in Programming Education: Easing the Burden on Beginners with REXX. In *Central European Conference on Information and Intelligent Systems* (pp. 171-178). Faculty of Organization and Informatics Varazdin.