第 2 章：選對用例的重要性

CHAPTER 2The Importance of Picking the Right Use Case

正如你在前一章的案例研究看到的，選錯用例往往是 AI
專案走向失敗的第一個不祥徵兆。這很常見。本章會提供一份挑選 AI
驅動專案正確用例的詳細指南——這是你成功的基礎。

As you saw in the case study in the previous chapter, picking the wrong use case is often the first harbinger of doom for the AI project. This is a common problem. This chapter provides a detailed guide for picking the right use case for your AI-driven project—the foundation of your success.

接下來的故事來自我曾為一家精準灌溉公司擔任顧問的經驗。他們想用 AI 告訴農民該在什麼時候、要澆多少水，才能讓作物得到足夠水分。許多潛在客戶是第三代農夫，他們的父親與祖父輩曾耕作同一片土地，如今由他們接手照料。這些農夫愛土地，就像愛自己的孩子一樣。

The following story comes from the time I was consulting for a precision irrigation company. This company was trying to use AI to tell farmers when and how much to water their crops so the plants were sufficiently watered. Many of their potential customers were third-generation farmers whose fathers and forefathers tilled that same land they were now caring for. Farmers who knew and loved their land the way they loved their children.

（你大概知道我要說什麼了.....）

(You see where I’m going with this…)

當我們做使用者研究時發現，這家公司想賣 AI
解決方案的那些農夫，其實完全不缺判斷土壤含水量的能力。他們不需要 AI、昂貴感測器或電腦模型。相反地，他們用一個流傳已久的「踢土測試」——名副其實：每天早上，農夫走進田裡，用靴子的腳跟往土裡一踢。若靴子能陷進土裡，瞧——作物的水分就夠了。

When my team did the user research, we discovered that farmers like the ones the company was trying to sell their AI solution to had no difficulty whatsoever determining adequate soil moisture levels. They did not need AI, fancy sensors, or computer models. Instead, they used a time-honored “kick test” performed precisely how it sounds: Every morning, the farmer goes into the field and kicks the heel of their boot into the soil. If their boot sinks into the soil, voilà—the crops are sufficiently watered.

就這樣。

And that was that.

事實上，要一位第三代農夫仰賴 AI 告訴他作物水夠不夠，根本就是在侮辱人。潛在客戶覺得這家 AI
公司自以為是，想教他們怎麼做生意。

In fact, the idea that a third-generation farmer would rely on AI to tell them that their crops were sufficiently watered was downright insulting. Potential customers felt that this AI company was presuming to tell them how to run their businesses.

以為 AI 會教專家怎麼做事，是一個紅旗警訊

Presuming That AI Will Be Telling Experts How to Do Their Job Is a Red Flag

就像上一章「義大利麵事件」的案例研究一樣，這個專案的目標是用 AI 取代一位受過訓練、經驗扎實的人類專家。這是失敗配方。只要你把「人類專家」和「機器」拿來正面對決推銷，就注定行不通——永

遠不會。事情就是這麼簡單。

Like the “spaghetti incident” case study in the previous chapter, this project aimed to replace a trained, established human expert with AI. This is a recipe for failure. Whenever you try to pitch a human expert versus a machine directly, it’s just not going to work—not ever. It’s as simple as that.

不管你有多少訓練資料集，或 SME（領域專家）寫了多少規則，總會有你沒想到的邊界案例，總會有只適用於某一台設備、某一塊田地、某一種土壤的特殊情境。更別提人類專家天生的懷疑、驕傲與偏見——面對一台自以為能指揮他做事的機器，這些情緒立刻會擋在專案前面，讓你徹底翻車。

No matter how many training datasets you have or how many rules your SMEs create, there will always be some edge case you have missed, some bespoke condition that applies only to this machine or this particular field or soil. Natural human suspicion, pride, and prejudice of a trained expert toward a machine that would presume to tell them what to do will immediately get in the way of your project, and your goose will be cooked.

我的簡單建議？去找別的用例。

My simple advice? Look for another use case.

這也是我在這個案子裡採取的做法。

Which is what I have done in this case.

問一個更好的問題

Ask a Better Question

當我判斷「用 AI 取代專家對作物水分的判斷」行不通時，我只是改去問同一批農夫：到底是什麼事情真的讓他們夜裡睡不著？（爆雷：不是作物水分夠不夠。）

When I determined that replacing the expert’s judgment of how well the crops were watered wouldn’t work, I simply asked those very same farmers what was really keeping them up at night. (Spoiler alert! It was not whether their crops were sufficiently watered.)

透過多次使用者訪談，我發現讓農夫焦慮的是更沉重的議題，例如淡水資源日漸枯竭；例如政府對用水祭出更嚴格的新規；例如氣候變遷帶來乾旱，正一點一滴把他們心愛的加州土地變成毫無生氣的荒漠。

Instead, I found out through multiple user interviews that what stressed these farmers out was more weighty considerations, like dwindling supplies of fresh water. Like new, much more stringent government regulations controlling water use. Like climate change, bringing with it dry conditions that were slowly but surely turning their beloved California land into a lifeless desert.

很快就能看出，正確的用例不是這個：

It was immediately apparent that the correct use case was not this:

而是這個：

Instead, it was this:

差別很細微，但結果證明它至關重要。

It was a subtle difference, but it turned out to be critical.

注意

NOTE

公司想賣的是一種「灌溉保險」，確保水量「足夠」。農夫想要的是：在不影響作物的前提下，自己到底能把用水量壓到多少。

The company wanted to sell a kind of “irrigation insurance” to make sure that sufficient water was applied. The farmers wanted to figure out how little water they could get away with applying.

對 AI 來說，真正值得投入火力的用例，是那些會讓客戶夜裡睡不著的問題。這個關鍵研究洞見，讓「以 AI 支援的精準灌溉」對我的客戶與他們的客戶都變成可行的策略。

The right use cases on which to focus the power of AI were the ones that kept our customers up at night. That key research insight about the right use case made AI-based precision irrigation a viable play for my client and their customers.

Jakob Nielsen 曾說過一句很有智慧的話：「如果你把望遠鏡對準土星，你會看到它有光環」（1）。把望遠鏡對準正確的方向（也就是問對問題），再提供足夠的放大倍率（以專注、同理與開放的心態訪談足夠多的人），你就會得到同樣的結果。

In the wise words of Jakob Nielsen, “If you point your telescope at Saturn, you will see that it has rings” (1). Point the telescope in the right direction (ask the right question), apply sufficient magnification (interview enough people with focus, empathy, and an open mind), and you will get the same result.

注意

NOTE

本質上，UX 是一門很直白的專業。其他領域的專家靠提供答案賺錢；而 UX 從業者在某種意義上，是靠販售自己的無知賺錢。換句話說，好的 UX 會問出好問題。

At its heart, UX is a straightforward discipline. Whereas other industry experts make money by providing answers, UXers, in essence, make money by peddling their ignorance. In other words, good UXers ask good questions.

既有的 UX 方法——例如情境式探查、田野研究、使用者訪談等等——最終都是在系統性地釐清客戶需求。接著，UXer 可以用親和圖（affinity mapping）、顧客旅程建模等工具，萃取出對 AI 專案來說價值連城的黃金洞見：可獲利的用例、潛在營收改善、新市場機會等等。

Existing UX methods, such as contextual inquiry, field studies, user interviews, and the like, all boil down to systematically determining customers’ needs. Then, UXers can use tools like affinity mapping and customer journey modeling to extract the nuggets of wisdom that are solid gold for an AI project: profitable use cases, potential revenue improvements, new market opportunities, and so on.

好消息是，這些歷久彌新的 UX 研究方法，就像伽利略的老望遠鏡一樣，用在 AI 專案上完全沒問題——你只要真的去用它們！

The good news is that time-honored UX research methods, like Galileo Galilei’s old telescopes, work just fine for AI projects—you just have to use them!

但如果你不問問題，反而倚賴那種自大的「藍海 / 紅海」鬼扯來決定 AI 用例，那你就跟在伸手不見五指的黑房間裡、戴著巨大的焊接手套去抓一隻黑貓沒兩樣。

However, if you fail to ask questions, if you instead rely on the arrogant “blue ocean/red ocean” bullshit to determine your AI use cases, you might as well be trying to catch a black cat in a pitch-black room while

wearing giant welding gloves.

你不會開心。

No fun for you.

貓也不會開心。

No fun for the cat.

沒有人會開心。

No fun for anyone.

注意

NOTE

只要你試圖把人類專家和機器拿來直接對打推銷，就一定行不通——永遠不會。事情就是這麼簡單。要開始把問題框對，你應該去問客戶：到底什麼才是真正讓他們夜裡睡不著？

Whenever you are trying to pitch a human expert versus a machine directly, it's just not going to work. Not ever. As simple as that. To begin framing the right problem, you should ask your customers about what's really keeping them up at night.

打對問題之所以關鍵，我把本書第一部整個都獻給了這個主題。下一章我會說明，如何用分鏡（storyboarding）協助你為 AI 專案挑出正確用例；第 4 章我會展示如何結合數位分身（digital twin）做深入的 UX 建模，確保你框對問題，而且方式也符合你的資料資源與 AI 能力；最後，第 5 章我會介紹一個新的 AI 模型 UX 評估方法「價值矩陣（value matrix）」，幫你在開始設計解法之前，把問題定義的每個面向都釘牢。

Attacking the right problem is so critical that I have devoted the entire Part 1 of this book to the subject. In the next chapter, I will explain how storyboarding can help you select the right use case for your AI project. Then, in Chapter 4, I will show you how to use in-depth UX modeling with a digital twin to ensure that you are framing the right problem and doing it in a way congruent with your data resources and AI capabilities. Finally, in Chapter 5, I'll introduce a novel AI model UX evaluation methodology called "value matrix" to help you nail every aspect of the problem definition before you begin designing the solution.

也別錯過 Thomas Wilson 的側欄〈全球醫療照護中有前景的 AI 用例〉。他描述的 10 個用例，可能很快就會出現在你附近的醫療機構。這些醫療用例很強大，但同時也是一把雙刃劍：它們結合了驚人的投資報酬率（ROI）與立即的病患效益，卻也帶來巨大的傷害與濫用風險。它們提醒我們 UXer 的責任：要與病患、醫師、護理師做形成性研究，聚焦在他們（往往非常不同，甚至互相對立）的需求上；同時也要設計隱私防護欄、讓人類介入的機制，以及在 AI 犯錯時（不是如果，而是何時）能讓使用更安全、把傷害降到最低的方法。

Be sure to check out the "Promising AI Use Cases in Global Healthcare" sidebar by Thomas Wilson. The 10 use cases Thomas describes might be coming soon to a healthcare facility near you. These healthcare use cases are powerful, yet they are also a double-edged sword, combining tremendous return on investment (ROI) and immediate patient benefits with a potential for great harm and misuse. These use cases underscore the work we must do as UXers: conduct formative research with patients, doctors, and nurses, focusing on their (often quite different, if not outright opposing) needs. Also, design guardrails and safeguards for patient privacy, mechanisms for human intervention, and ways to make the AI use safer and minimize potential harm when (not if) the AI makes mistakes.

另一個重點是：不是所有用例都適合用 AI 來支援。想深入這個重要議題，請讀 David Andrzejewski 的側欄〈選對 AI/ML 用例〉。下一章我們會討論：要如何判斷一個心理健康助理 App 的用例好不好。也鼓勵你用同樣的思路，思考本書介紹的各種技巧，如何幫助這 10 個複雜的醫療用例在安全與效能上都更到位：什麼條件會讓某個用例成為（或不是）AI 支援的好候選？AI 的影響力應該延伸到多遠？用 AI 解決這些問題可能帶來哪些效益與風險？在新的常態下，協助團隊找出答案，是 UX 專業人士的責任。

Another key point is that not all use cases are good candidates for AI support. To dive deeply into this important topic, read David Andrzejewski's sidebar, "Selecting the Right AI/ML Use Case." In the next chapter, we will discuss how to determine what makes for a good or bad use case for a mental health assistant app. We encourage you to take the same approach to reflect on how various techniques described in this book can help ensure the safety and efficacy of AI in these 10 complex healthcare use cases. What would make a particular use case a good or bad candidate for AI support? How far should AI extend its influence? What might be some potential benefits and dangers of using AI to solve these problems? In the new normal, it is up to UX professionals to help their teams figure out the answers.

參考資料

Reference

1. 1. Nielsen, J.（2007）。〈橫幅視而不見：最初的眼動追蹤研究〉。NNGroup。www.nngroup.com/articles/banner-blindness-original-eyetracking

1. 1. Nielsen, J. (2007). Banner blindness: The original eyetracking research. NNGroup. www.nngroup.com/articles/banner-blindness-original-eyetracking

觀點：全球醫療照護中有前景的 AI 用例

PERSPECTIVE: PROMISING AI USE CASES IN GLOBAL HEALTHCARE

作者：Thomas Wilson

By Thomas Wilson

以下列出 10 項強大的 AI 驅動創新，聚焦在醫療照護：

Here are 10 powerful AI-driven innovations with a focus on healthcare:

1. 藥物研發中的生成式 AI：像 ChatGPT、DALL·E 這類工具能透過生成可供測試的新化合物，加速藥物發現。在醫療客服中心，AI 也能自動草擬針對病患詢問的個人化回覆，加快回應速度並提升滿意度。

1. Generative AI in Drug Discovery: AI tools like ChatGPT and DALL-E accelerate drug discovery by generating novel compounds for testing. In healthcare call centers, AI can automatically draft personalized responses to patient inquiries, speeding up response times and improving patient satisfaction.

2. 個人化醫療中的 AI：醫療平台上的預測模型能依個人基因資訊制定高度個人化的治療計畫。AI 也能協助醫療客服中心分流病患需求，並更有效地把緊急個案轉給專門團隊。

2. AI in Personalized Medicine: Predictive models in healthcare platforms allow for highly personalized treatment plans based on individual genetic information. AI also assists healthcare call centers in triaging patient concerns and routing critical cases to specialized teams more effectively.

3. 健康平台的 AutoML：AutoML 平台正幫助醫院與醫療系統在不需要龐大資料科學團隊的情況下部署機器學習模型。這能透過預測住院人數與資源需求來改善營運。在客服中心，AutoML 可自動化掛號排程、追蹤照護指示等客服流程。

3. AutoML for Health Platforms: AutoML platforms are helping hospitals and healthcare systems deploy machine learning models without needing large data science teams. This technology improves operations by predicting patient admissions and resource needs. In healthcare call centers, AutoML automates customer service processes such as appointment scheduling and follow-up care instructions.

4. 可解釋的醫療診斷 AI：在醫療領域，可解釋 AI（XAI）能讓決策更透明，特別是在診斷上。例如 AI 可以說明為什麼某些症狀指向特定疾病，增進醫師與病患的信任。在醫療平台上，這也能幫助客服人員用更易懂的方式向病患解釋健康資訊。

4. Explainable AI for Medical Diagnostics: In healthcare, explainable AI (XAI) ensures transparency in decision-making, particularly in diagnostics. For instance, AI can explain why certain symptoms point to specific diseases, improving trust between doctors and patients. In healthcare platforms, this helps call center representatives give patients more understandable information about their health.

5. 遠距醫療與客服中心的 NLP：自然語言處理（NLP）讓醫療平台能自動轉錄並分析醫病對話，更容易追蹤病患關切。NLP 也能強化客服中心的情緒分析，使座席能即時理解病患情緒並調整回應。現在我們也能使用 Google Translate API，確保所有病患都能用母語獲得協助。

5. NLP in Telemedicine and Call Centers: Natural language processing (NLP) enables healthcare platforms to automatically transcribe and analyze patient–doctor conversations, making it easier to track patient concerns. NLP enhances sentiment analysis in call centers, allowing agents to understand patient emotions in real time and adjust their responses accordingly. We currently have the ability to use Google Translate APIs to ensure that all patients receive assistance in their native tongue.

6. 醫療裝置的 Edge AI：邊緣 AI 讓醫療裝置能更自主運作，例如可攜式超音波或健康監測穿戴裝置。對客服中心而言，edge AI 能提供即時診斷支援，降低人工諮詢需求。

6. Edge AI in Medical Devices: Edge AI powers medical devices that can operate autonomously, such as portable ultrasound machines or health monitoring wearables. For call centers, edge AI can provide real-time diagnostic support, reducing the need for manual consultations.

7. 醫療資安中的 AI：隨著健康紀錄日益數位化，AI 驅動的安全系統能保護病患資料免於外洩。在醫療平台與客服中心，AI 會監控可疑活動，防止未授權存取敏感資訊。

7. AI in Healthcare Cybersecurity: With the increasing digitization of health records, AI-driven security systems protect patient data from breaches. In health platforms and call centers, AI monitors for suspicious activity and prevents unauthorized access to sensitive patient information.

8. 健康系統的 MLOps：MLOps 簡化醫院與診所部署機器學習模型的流程，讓資源管理更有效率，也能做設備維護的預測。客服中心也能藉由 MLOps 整合 AI 模型來預測病患滿意度或等待時間。

8. MLOps for Health Systems: MLOps streamlines the deployment of machine learning models in hospitals and clinics, enabling efficient resource management and predictive equipment maintenance. Healthcare call centers benefit by using MLOps to integrate AI models that predict patient satisfaction or wait times.

9. HealthTech 的低碼 / 無碼 AI：HealthTech 新創與醫院如今能利用低碼 AI 平台，快速開發病患管理、詐欺偵測、健康結果預測等應用，而不需高深技術專長。在客服中心，這些平台可快速部署新工具，處理病患問題並簡化工作流程。我們甚至非常接近讓網站與軟體不再需要傳統

UI，而改由禮賓式的機器人代理，提供個人化體驗。

9. Low-Code/No-Code AI for HealthTech: HealthTech startups and hospitals can now leverage low-code AI platforms to quickly develop applications for patient management, fraud detection, and health outcome prediction without needing advanced technical expertise. In call centers, these platforms allow for the rapid deployment of new tools to handle patient queries and streamline workflows. We are very close to not even having traditional UI in websites and software, but instead, a concierge bot agent to serve you with a personalized experience.

10. 醫療員工賦能的 AI：協作式 AI
能強化醫護人員訓練，依表現提供個人化學習模組。這也反映在客服中心：AI
工具可為座席提供即時支援，提升他們快速且正確回答複雜醫療問題的能力。

10. AI for Employee Enablement in Healthcare: Collaborative AI in healthcare enhances the training of medical staff, offering personalized learning modules based on their performance. This is mirrored in healthcare call centers, where AI tools provide real-time support for agents, improving their ability to answer complex medical queries quickly and accurately.

這些 AI 與 ML 應用正從病患照護到醫療客服中心與平台的幕後作業，全面改變醫療營運，並在整體上提升效率與照護成果。

These AI and ML applications are transforming healthcare operations, from patient care to the behind-the-scenes work of healthcare call centers and health platforms, improving efficiency and care outcomes across the board.

關於 Thomas Wilson

About Thomas Wilson

Thomas Wilson 是得獎的 UX、CX、EX、服務設計師、組織設計師與設計總監，目前在醫療領域提供旅程管理與策略領導。他曾推動 53 家新創的關鍵進展，並為 United Healthcare、BCBS、Tenet、HCA、AIG、Experian、AWS、NASA、Kroger 等客戶帶來創新與轉型；同時也服務眾多科技新創、中小企業（SMB）與《財富》前 5 / 前 500 企業，涵蓋 AI、大數據、金融科技、醫療、零售與資安等領域。可透過 www.linkedin.com/in/thomasianwilson 聯絡他。

Thomas Wilson is an award-winning UX, CX, EX, Service Designer, Organizational Designer, and Design Director. He currently provides journey management and strategic leadership in healthcare. Thomas has moved the needle at 53 start-ups and has innovated and transformed clients such as United Healthcare, BCBS, Tenet, HCA, AIG, Experian, AWS, NASA, and Kroger as well as a host of tech start-ups, small and medium-sized businesses (SMBs), and Fortune 5s/500s in AI, big data, financial technology, healthcare, retail, and security. He can be reached at www.linkedin.com/in/thomasianwilson.

觀點：選擇正確的 AI/ML 用例

PERSPECTIVE: SELECTING THE RIGHT AI/ML USE CASE

作者：David Andrzejewski

By David Andrzejewski

使用者越來越期待現代軟體除了基本功能之外，還能展現更聰明的行為。那種 App 彷彿「神奇地」推薦你可能想看的影集、想吃的食物或想買的商品所帶來的愉悅感，和你必須辛苦地一一指定需求的體驗差

異，可能非常巨大。如果你不願意投入打造這種體驗，你的競爭對手一定會。實務上，這種「魔法」多半由機器學習（ML）或人工智慧（AI）技術提供。這些系統純粹技術面的部分，本身就是一個迷人且快速演進的領域；但在這裡，我們會聚焦在它們對產品設計與使用者體驗（UX）的影響：我們該如何把 AI/ML 能力整合進產品體驗中，讓風險與挑戰降到最低，同時讓使用者獲得最大好處？

Users increasingly expect modern software applications to deliver intelligent behaviors above and beyond basic functionality. The subjective difference between the delight of using an app that magically recommends what you might want to watch, eat, or purchase versus one where you have to laboriously specify your desires can be huge. If you aren't willing to invest in making that happen, your competitors surely are. In practice, this magic is delivered via machine learning (ML) or artificial intelligence (AI) techniques. The purely technical aspects of these systems form a fascinating and fast-moving field in their own right. Still, here, we will focus on the product design and user experience (UX) implications of these technologies. How can we integrate AI/ML capabilities into a product experience in such a way that the risks and challenges are minimized while our users reap the maximum benefits?

AI/ML 的優勢之一，是能為那些難以或不可能用「傳統」軟體手工打造解法的問題提供答案，並處理本質上嘈雜、帶機率性的領域。以手寫數字辨識為例：如果沒有
AI/ML，你很可能會寫出一堆奇怪又互相重疊的規則，等於用更差的方式重新發明
AI/ML。從體驗角度來看，AI/ML 的一大缺點是輸出缺乏確定性或校準（calibration）。所謂確定性，是指對於任一輸入，你無法事先知道輸出品質；同一次互動結果可能好也可能壞，而在產生結果之前我們通常無從得知；也無法把壞結果的機率壓到零。所謂校準，是指系統往往無法偵測或描述自己對結果的信心：它並不知道輸出好不好，甚至可能「自信地答錯」。因此，系統設計者真正有趣（也最難）的地方，是思考如何在不可靠、也不一致的地基上，打造可靠且一致的使用者體驗。

Some advantages of AI/ML are its ability to provide answers to problems for which it is difficult or impossible to hand-craft "classic" software solutions and to deal with intrinsically noisy or probabilistic domains. Consider the problem of handwritten digit recognition; in the absence of AI/ML, one would likely write a bunch of strange overlapping rules, essentially reinventing the AI/ML approach poorly. One major downside of AI/ML approaches, especially from an experience perspective, is the lack of determinism or calibration in its outputs. By determinism, we mean that, for any given input, the quality of the resulting output cannot be known in advance. For a given user interaction, the result may be good or bad, but we generally do not know for sure which it will be until we generate the result, and we cannot drive the probability of bad results down to zero. By calibration, we mean that, by default, the system itself is often unable to detect or qualify the confidence in the result. That is, it does not "know" whether or not its output is good and may be "confidently wrong." Therefore, the fun part for system designers is to consider how we might try to build reliable and consistent user experiences atop unreliable and inconsistent foundations.

產品 / 技術選擇：最好的 AI 是沒有 AI

Product-Technical Choices: The Best AI Is No AI

下面這句話看起來像是顯而易見，但熱情的建造者有時會忘記：使用者通常「不想要」AI；他們想要的是達成特定結果或解決特定問題。你用什麼機器來交付，對使用者而言只是手段。如果有一種直接、可預期地提供高品質結果的方法，不需要動用
AI/ML，那就大方去做吧，然後一邊笑著進帳，一邊看同行還在排查 Apache Airflow
資料管線或除錯各種玄學 PyTorch 錯誤。

The following statement may seem like a straightforward point. Still, it is one enthusiastic builders can sometimes forget: Users, in general, do not "want" AI but instead want to achieve their particular outcome or solve their specific problem. The machinery by which you deliver this is, to the user, simply a means to their ends. Suppose a straightforward way to deterministically provide high-quality results exists without resorting to AI/ML. Then, by all means, do that and laugh all the way to the bank while your peers troubleshoot Apache

Airflow data pipelines or debug esoteric PyTorch errors.

次好的 AI，是「無聊的 AI」

The Next Best AI Is “Boring AI”

即使必須用 AI/ML，也可能有好處去選擇更簡單、較「經典」的方法，例如決策樹或線性迴歸，而不是追最新、上週才丟到 arXiv 的前沿技術。至少，這些較直觀（甚至是非 AI/ML 的規則式）方法應該要先實作並當作基準線，用來評估你後續 AI/ML 系統到底有沒有帶來價值（或根本沒有）。你的語意向量嵌入檢索系統在這個問題上真的比 BM25（Best Match 25）更好嗎？只有一種方法能知道！

Even if AI/ML is required, there can also be benefits to reaching for simpler “classic” methods like decision trees or linear regression instead of bleeding-edge techniques that were posted to arXiv earlier this week. At the very least, these more straightforward (or even non-AI/ML rule-based) approaches should be implemented and considered as baselines to judge the value add (or lack thereof) of your later AI/ML-powered system. Is your semantic embedding vector retrieval system better for this problem than BM25 (Best Match 25)? There’s only one way to find out!

把你的問題對應到一個大家熟悉的 AI/ML 問題

Map Your Problem to a Well-Understood AI/ML Problem

熟悉某個特定 AI/ML 演算法其實沒那麼重要；反而，熟悉 AI/ML 的「問題設定」常被低估。對多數實務應用而言，針對聚類問題選 X 模型還是 Y 模型，影響往往遠不如你能否意識到：你真正面對的是一個排序（ranking）問題。務必讓懂得核心 AI/ML 問題設定（分類、迴歸等等）的人，深入思考客戶的具體用例，並把它框成一個良好定義、可解的 AI/ML 問題。

Knowing specific AI/ML algorithms is overrated, whereas knowing AI/ML problem settings is underrated. For many practical applications, choosing model X over model Y for a clustering problem will be dramatically less impactful than realizing that you have a ranking problem. It is essential to have someone who understands the handful of core AI/ML problem settings (classification, regression, etc.) think deeply about the specific customer use case and figure out how to frame the issue as a well-posed AI/ML problem.

採用最佳實務

Use Best Practices

如前所述，AI/ML 驅動系統的技術設計與實作仍是一門新興專業。借鑑社群的經驗、最佳實務與設計模式很合理。例如，你應該仔細策劃高品質的訓練資料集、嚴格評估模型表現，並在上線系統中密切監控關鍵績效指標（KPI）。

As mentioned earlier, the technical design and implementation of AI/ML-driven systems is an emerging discipline. It certainly makes sense to use the lessons learned, best practices, and design patterns from those communities. For example, you should carefully curate high-quality training datasets, rigorously evaluate model performance, and carefully monitor crucial key performance indicators (KPIs) in the deployed system.

UX 的選擇

UX Choices

假設我們已確定 AI/ML 的確是正確工具，並且如前所述把核心技術路徑的風險降到最低，那麼接下來要怎麼調整整體體驗，才能最大化好處、同時降低風險？

Assuming that AI/ML techniques are truly the right tool for the job and that we have maximally de-risked the core technological approach, as described earlier, how can we adjust the rest of the experience to maximize the benefits and mitigate the risks?

有韌性的核心

Resilient Core

你的 App 核心功能應該能承受 AI/ML 結果品質不佳的情況。例如，推薦、預填、以及自動完成在做對時能省你時間，但做錯時不該把你整個卡死。照片編輯軟體的自動選取也是如此：如果效果不好，你仍然可以用傳統方式手動選取。相反地，如果整個體驗都是「只靠 AI」，而 AI 又不照你想的做，會讓人抓狂。

The core functionality of your app should be resilient to poor-quality AI/ML results. For example, recommendations, pre-fill, and autocomplete save you time if they're good, but don't block you entirely if they're bad. In photo-editing software, auto-selection also has this property: If it isn't working well, you can still select the old-fashioned way. Alternatively, an "AI only" experience could be maddening if the AI isn't doing what you want.

讓使用者幫助你

Help the User Help You

使用者可以透過部分表達意圖或提供其他提示，讓 AI/ML 任務變得更容易。例如，有些分面搜尋介面會用篩選條件把後續 AI/ML 搜尋可用資料集確定性地縮小到某個子集。另一個有趣例子是串流媒體的「誰在看？」個人檔案選擇畫面。理論上，你可以用很複雜的 AI/ML 從觀看資料自動學出不同使用者檔案，並預測現在是哪個人正在使用；但直接問使用者更簡單也更可靠。

The user can make the AI/ML task easier by partially conveying intent or giving other hints. For example, there are some faceted search interfaces where the facet filters deterministically restrict the subsequent AI/ML search to some subset of the dataset. Another fun example is the "Who's Watching?" profile selection screen on streaming media services. Hypothetically, it may be possible to use sophisticated AI/ML to automatically learn distinct profiles from viewing data and predict which profile currently uses the service. Still, simply asking the user is much simpler and more reliable.

用商業邏輯包住 AI/ML

Wrap AI/ML in Business Logic

不要害怕用簡單的商業邏輯程式碼來處理防護欄或邊界案例。想像一個廣告競價系統用 AI/ML 來決定某次曝光的最佳出價。不管你的 AI/ML 工程師多厲害，若不先套上商業邏輯去限制出價與花費在合理範圍內，就把模型輸出直接丟進下游的下單執行系統，都是不明智的。同樣地，在面向使用者的 App 上，你也可能需要用硬編碼的商業規則來處理特定邊界情境或避免特定失敗模式。

Don’t be afraid to use simple business logic code for guardrails or edge cases. Consider an ad bidding system that uses AI/ML to determine the optimal price to bid for a given impression. No matter how skilled your AI/ML engineers are, it would be inadvisable to propagate the model output directly into the downstream bid order execution system without applying some business logic to ensure bid prices, and spending is bounded by reasonable limits. Likewise, for a user-facing app, you may want to wrap your AI/ML outputs in hard-coded business logic rules to handle certain edge cases or avoid particular failure modes.

校準與可解釋性

Calibration and Explainability

校準與可解釋性需要在建模層面投入額外成本，但能大幅改善 App 在 AI/ML 面向的使用者體驗。校準在不同情境下意思略有差異，但本質上它涵蓋的是對「預測本身」的二階知識。粗略來說，它可以轉譯成：某個輸出有多可能正確或相關；真實值以某個機率落在某個範圍或集合內的保證；或直接輸出明確且有意義的機率。如何在 UX 中使用或呈現這些資訊仍是個有趣的開放問題。一個簡單做法是：當 AI/ML 低信心時回覆「不知道」，或退回確定性的基準方法。

Calibration and explainability require additional investment at the technical modeling level but enable significantly improved user experience on the AI/ML aspects of an app. Calibration can mean subtly different things in different contexts, but fundamentally, it covers second-order knowledge about the predictions themselves. Loosely, this can translate into claims about how likely a given output is accurate or relevant, guarantees that the true value lies in some range or set with some probability, or gives explicit and meaningful probabilities as outputs. An interesting open question is how to use or express this information in the UX. A simple approach might be to answer “Don’t Know” or fall back to deterministic baselines when the AI/ML gives low-confidence answers.

可解釋性也涵蓋各式各樣的方法，通常指的是用來判定某個 AI/ML 系統為什麼、或如何得到某個輸出的技術。例子包括：大型語言模型（LLM）聊天機器人提供引用來源，或在串流推薦旁加上「因為你喜歡 X，所以推薦」之類的說明。這些額外脈絡能讓使用者更容易判斷輸出的可信度或相關性，進而提升 AI/ML 能力的實用性。

Explainability also covers a variety of methods and generally refers to techniques for determining why or how a given AI/ML system has arrived at its output. Examples include providing citations in a large language model (LLM) chatbot or supplying “recommended because you liked X” alongside streaming media recommendations. This additional context can enhance the usefulness of AI/ML capabilities by helping the user judge the credibility or relevance of the outputs.

人類心理與演算法

Human Psychology and Algorithms

AI/ML 應用另一個令人興奮的發展，是社會科學研究開始深入探討：使用者如何看待由演算法或模型驅動的軟體系統輸出。這個領域進展快速，並可能帶來可落地的洞見。一些初步發現聚焦在「演算法厭惡」（algorithm aversion）（1）：人們通常對演算法系統（例如 AI/ML 模型）的差結果容忍度，比對同樣糟的「人類」結果更低。後續研究發現，如果讓終端使用者有一些能力去調整或修改 AI/ML 輸出，或展示 AI/ML 系統表現正在改善，這種效應可以被削弱（2）。把這些元素納入 UX 設計，可能有助於提升使用者對演算法能力的滿意度。

Another exciting development for AI/ML applications is the rise of social science research on how users perceive the outputs of algorithm or model-driven software systems. This is a fast-moving field with potentially actionable insights. Some interesting preliminary findings center around algorithm aversion (1). In

this phenomenon, people are generally less tolerant of poor results from algorithmic systems (such as AI/ML models) than they would be if equally poor results were delivered by a human. Follow-up research found that this effect could be attenuated by giving end users some ability to adjust or modify AI/ML output or demonstrating that the AI/ML system performance is improving (2). Incorporating such elements in your UX design may improve user satisfaction with algorithmic capabilities.

結論

Conclusion

使用者與市場對軟體應用的要求只會越來越強大、越來越直覺。AI/ML 技術提供了一套有價值的工具箱來滿足這些需求，但新工具也帶來新挑戰。打造應用程式是一項團隊運動；要真正發揮 AI/ML 的力量，同時馴服它的缺點，需要在產品設計與使用者體驗上做出相應調整。這是一個令人興奮、而且快速演進的領域。希望這些策略能為你踏上 UX for AI/ML 的旅程提供一些起點。

Users and the marketplace demand ever more powerful and intuitive software applications. AI/ML technologies provide a valuable toolkit for delivering on these requirements, but new tools come with new challenges. Building applications is a team sport, and it requires product design and user experience adaptations to fully harness the power of AI/ML while taming its downsides. This is an exciting and rapidly evolving space in which to work. Hopefully, these tactics can provide some starting points for your journey in UX for AI/ML!

參考資料

References

1. 1. Dietvorst, B. J., Simmons, J. P., & Massey, C.（2018）。〈克服演算法厭惡：如果能（哪怕只是一點點）修改，人們也願意使用不完美的演算法〉。Informs。https://faculty.wharton.upenn.edu/wp-content/uploads/2016/08/Dietvorst-Simmons-Massey-2018.pdf

1. 1. Dietvorst, B. J., Simmons, J. P., & Massey, C. (2018). Overcoming algorithm aversion: People will use imperfect algorithms if they can (even slightly) modify them. Informs. https://faculty.wharton.upenn.edu/wp-content/uploads/2016/08/Dietvorst-Simmons-Massey-2018.pdf

2. 2. Berger, B., Adam, M., Rühr, A., & Benlian, A.（2020）。〈看我變更好——演算法厭惡與展示學習能力〉。Springer Nature。https://link.springer.com/article/10.1007/s12599-020-00678-5

2. 2. Berger, B., Adam, M., Rühr, A., & Benlian, A. (2020). Watch me improve—Algorithm aversion and demonstrating the ability to learn. Springer Nature. https://link.springer.com/article/10.1007/s12599-020-00678-5

關於 David Andrzejewski

About David Andrzejewski

David Andrzejewski（www.david-andrzejewski.com）是位於舊金山的應用型機器學習專業人士。最近他擔任 Sumo Logic 的 AI Experiences 工程總監，專注於現代軟體應用的可靠性與安全相關議題。更早之前，他在完成威斯康辛大學麥迪遜分校電腦科學博士後，於勞倫斯利佛摩國家實驗室（LLNL）以博士後研究員身分研究知識發現。

David Andrzejewski (www.david-andrzejewski.com) is an applied machine learning professional based in San Francisco. Most recently, he worked as Director of Engineering for AI Experiences at Sumo Logic, focused on problems related to the reliability and security of modern software applications. Previously, he investigated knowledge discovery at Lawrence Livermore National Laboratory (LLNL) as a postdoctoral researcher after completing his PhD in computer sciences at the University of Wisconsin–Madison.