從「宇宙能量」到 **INSUFFICIENT EVIDENCE**：我如何逐漸學會把抽象裝上 **TYPE SYSTEM** 🧠📚

關於寫作、數學、科學與可更新性的隨筆

DORIAN VERLAINE

Abstract. 本文回顧我如何在寫作、數學與物理的訓練中，逐漸把無邊界的抽象改造成可定義、可展開、可檢查，也可更新的思考模型。從「具體來說？」這個最小 debugger 出發，文章依次討論論證結構、前提與證據、經驗與權威、可證偽性、跨背景溝通，以及何時應該停止一場沒有收斂條件的對話。核心主張不是拒絕抽象或直覺，而是要求抽象有 schema、直覺標註 confidence，並讓自己的模型始終保留 update path。

有一類文字很容易讓人產生一種奇妙的感覺：它說了很多，似乎每一句都很有智慧，最後卻很難回答一個最基本的問題，它到底說了什麼？例如「提升能量」「回到真我」「理解宇宙的頻率」「完成生命課題」「進入更高維度的覺知」。這些句子在語感上往往很順，甚至帶有一種「好像碰到了某個很深的東西」的質感，但只要追問一句「具體來說呢？」，整個系統就可能開始掉封包。🤡

這次對話最開始只是從一本 self-help 書出發。我本來只是獵奇，想看看這種把「處世之道、心理成長、系統思維、東方哲學、宇宙能量」裝進同一本書裡的東西到底能長成什麼樣。結果一路聊下去，我慢慢發現，真正有趣的問題其實不是那本書本身，而是另一件更廣泛的事：為什麼人會把智慧包裝成一種越來越抽象、越來越難檢查、甚至連自己都不一定知道自己在說什麼的結構？以及，我自己是怎麼逐漸從這種狀態裡走出來的？

1. 抽象本身沒有錯，失去邊界的抽象才有問題

我以前也很容易滑進一種「無邊界抽象」的狀態。看到幾件具體的事情之後，腦子會迅速開始壓縮：

友情 → 關係 → 權力 → 結構 → 系統 → 存在 → 宇宙 🤡

這種壓縮其實很誘人，因為它給人一種「我正在看到別人沒看到的共同結構」的感覺。問題在於，當抽象能力先長出來，而驗證機制還沒跟上的時候，很多東西只剩下「整體感」，卻沒有足夠清楚的定義、例子、反例和中間推理。於是自己會覺得「我明明想到了一個很大的東西，為什麼別人聽不懂」，但真的要指出對方到底哪裡沒懂，自己也說不清楚。

這時候最容易出現一種危險的替代：把「別人無法重建我的想法」理解成「別人理解能力不夠」。其實很多時候，更接近真相的說法是：我腦中有一團高度壓縮的 **semantic blob**，但我還沒有把它 **factorize** 成一條別人可以跟著走的路徑。

真正好的抽象不是逃離具體，而是對具體施加更嚴格的約束。數學之所以抽象，不是因為它可以隨便飄，而是因為它有 type system。你說向量空間，就要滿足一組公理；你說線性映射，就要保留加法和數乘；你說極限，就不能永遠停留在「越來越接近」這種感覺上，最後總要回到 ε 和 δ。抽象在這裡不是霧，而是壓縮格式，而且可以解壓。

很多「玄」的表達真正的問題也不在於它抽象，而在於它抽象但不受約束。一句「你現在的能量不對」，如果追問之後只是得到「就是你的狀態不好」，那它其實只是把一個普通概念換了一件宇宙皮膚。如果再進一步說「相同頻率的人會互相吸引，所以宇宙會把同頻的人送到你身邊」，那就更有意思了，原本只是比喻的「頻率」已經偷偷從 metaphor 升級成 mechanism。比喻 root 了整台機器。💀

因此，我現在很喜歡一個極其簡單的 debugger：

具體來說？

Date: 2026-09-11.
2020 *Mathematics Subject Classification.* Epistemology, Reasoning & Technical Metaphor (Essay).
Key words and phrases. epistemology, abstraction, type-systems, reasoning, communication.

「提升自己的能量。」

具體來說？

「規律作息、運動、減少讓你長期疲憊的人際關係。」

很好，這句終於落地了。

如果回答仍然是「讓自己的頻率回到更高的狀態」，那就繼續問：

> 具體來說？

如果遞迴十層還碰不到 observable reality，那至少可以標一個：

```
meaning: unresolved
confidence: low
```

這裡還有一個我後來才慢慢意識到的區別：壓縮和模糊並不是同一件事。好的抽象確實可以把大量細節壓縮成很短的表達，但這種壓縮應該是可逆的。只要有需要，我可以沿著定義、推理和例子把它重新展開。數學符號看起來高度壓縮，卻不是含糊；一個定理可能只有一行，但背後每個符號都有固定接口。相反，「生命能量」「高維覺知」之類的詞有時候看起來也很濃縮，卻找不到穩定的解壓算法。它們不是 compression，而比較像把解析度調低之後再宣稱畫面變得更深。🤡

所以現在遇到一個「很有道理」的抽象，我會多問一個問題：它能不能被展開？如果一個概念可以從一句話展開成定義、條件、例子、反例，再重新壓縮回原句，那它大概真的承載了資訊；如果一展開就只剩下另一組同樣模糊的詞，那它承載的可能主要是氣氛。

2. IELTS 寫作意外地改造了我的思維方式

真正讓我逐漸意識到這些問題的，並不只有數學。IELTS 寫作對我整個表達方式的影響，比我一開始預想的大得多。它表面上是在訓練英文，實際上卻像一套很殘酷的認知衛生系統：你到底在主張什麼？這一段的中心是什麼？這句和上一句到底是因果、轉折、補充還是舉例？你寫了 `therefore`，前面真的能推出後面嗎？你舉的例子是在支持論點，還是在旁邊表演？

練久了之後，大腦裡會自然長出一個小型 reviewer：

> 「這句很漂亮。」
> 「然後呢？」
> 「……」
> 「刪掉。」🤡

IELTS 還逼著我建立了「讀者模型」。以前表達很容易默認一件事：我腦中這幾個概念顯然連在一起，所以你應該懂。但讀者沒有 SSH 登進我的腦子。我要把中間那些 packet 發出去，要把隱含關係顯式化。A 為什麼導致 B，B 在什麼條件下導致 C，哪一步只是相關而不是因果，哪個例子只是 illustration 而不是 evidence，這些都不能靠讀者自己猜。

這件事和後來學數學居然形成了很奇妙的呼應。數學說：「不要跳步。」寫作說：「不要讓讀者猜你跳了什麼。」前者要求 proof structure，後者要求 argument structure，兩邊都在打擊同一種句子：

> 「我覺得它們之間有一種很深的聯繫。」

老師：什麼聯繫？

我：就是那種……

老師：

```
Band 6.0
```

🤡

另外一個很大的變化，是我開始越來越不相信「大詞本身等於深度」。`profound`、`fundamental`、`essence`、`humanity`、`civilisation` 這些詞並不是不能用，但一句話的 scope 越大，它需要承擔的證明責任也越大。「社交媒體摧毀了現代人的真實連結」聽起來很有氣勢，但「高頻使用短影音可能降低某些情境下維持長時間注意的能力」雖然不炸裂，卻終於可以被討論、檢驗和反駁。願意縮小句子，換取可驗證性，是一種很重要的成熟。

IELTS 對我另一個很持久的影響，是讓我開始區分「我有很多想法」和「我有一個論證」。前者只要求腦內有很多節點，後者還要求邊存在，而且方向不能亂接。很多看似資訊量很大的表達，其實只是把十個 related words 放在同一段裡，讀者能感覺它們屬於同一片雲，卻找不到哪一條邊是因果、哪一條是例證、

哪一條只是聯想。真正清楚的寫作則像把 graph 顯式畫出來：這是主張，這是理由，這裡有限定條件，這個例子只支持其中一部分。從那以後，我開始很難再滿足於「大概就是這個意思」。

3. LADR、Spivak、Stewart、Feynman：抽象開始有 schema 了

接觸 LADR、Spivak、Stewart，甚至《費曼物理學講義》之後，這種變化變得更明顯。它們最重要的影響之一，不只是「我學到了某些知識」，而是讓我第一次很強烈地感受到：真正嚴肅的抽象是有邊界的。

Spivak 不會允許我永遠用「越來越接近」代替極限的嚴格定義。LADR 不會允許我用一句「線性就是保持某種結構」混過去，它會逼我說清楚：什麼結構？保留哪些運算？具體滿足哪些性質？Feynman 的文字雖然直覺感很強，但它仍然不斷把概念拉回物理情境、量綱和可觀察結果。它們都在做同一件事：給抽象綁上鉛塊，防止它飛進平流層。

我後來越來越喜歡把這個轉變描述成：

以前：

```
idea: any
```

後來：

```
definition -> proposition -> argument/proof -> example -> counterexample
```

抽象終於有 type system 了。從此再看到一句「真正的智慧，是在更高維度上理解生命的本質」，腦子不太容易先出現 🤯，而是先出現：

```
define("wisdom")
define("dimension")
define("understand")
define("essence")
```

如果四個都 undefined，那它至少暫時還不是「深」，只是「沒有 schema」。

《費曼物理學講義》給我的另一個提醒則很有意思。第一卷第一章開頭，Feynman 在談科學時寫了一句非常短的話：**“the sole test of its validity is experiment.”**[1] 這句話當然不能粗暴地拿去處理所有哲學、倫理或價值問題，但對於那些聲稱自己在描述「世界實際如何運作」的命題，它是一個極好的防火牆。你可以從任何地方得到一個想法，可以靠直覺、比喻、夢、歷史經驗，甚至靠一個非常漂亮的故事，但只要它開始聲稱某種可觀察的因果關係，就必須準備和世界對帳。

Feynman 在同一章還做了一個很漂亮的思想實驗：如果科學知識遭到毀滅，只能留下一句話，他選擇原子假說，也就是世界由運動著、彼此作用的原子構成。[2] 我很喜歡這一段，不只是因為它講原子，而是因為它展示了一種和「玄」完全相反的高密度表達：一句話可以極度抽象、極度壓縮，卻能一路向下展開成水的性質、溫度、壓力、物質結構和大量可觀察現象。真正強的抽象不是讓你停止追問，而是讓你得到更多可以追問的東西。

這也讓我逐漸形成一個判準：如果一句「深刻」的話越解釋，世界變得越具體、預測越多、邊界越清楚，那它可能真的在壓縮知識；如果越解釋只會得到更多「能量、頻率、覺知、宇宙、合一」，那可能只是語義在遞迴呼叫自己。💀

4. 但問題來了：和沒有相同訓練背景的人交流，真的很累

這種變化當然也帶來副作用。當我開始習慣比較清楚地區分概念、檢查中間步驟、關心證據和抽象損失之後，和沒有經歷過類似訓練的人交流，有時候會變得很耗能。

比如我很習慣用計算機詞彙做類比。我可能會說「這比較像 cache miss，而不是能力差」。對我來說，這句話背後已經壓縮了很多前置內容：cache 是什麼，hit 和 miss 有什麼差別，為什麼第一次重算成本高，為什麼已有模型會大幅降低後續處理成本。但如果對方沒有這些背景，我實際上不是只說了一句話，而是偷偷依賴了 4 GB shared library。🤡

---

[1] Richard P. Feynman, Robert B. Leighton, Matthew Sands, *The Feynman Lectures on Physics*, Vol. I, Ch. 1, “Atoms in Motion”. Caltech online edition.

[2] 同上。Feynman 在 §1–2 用「如果只能留下一句科學知識」的思想實驗引出 atomic hypothesis，並用它展示高度壓縮的物理觀念如何展開成大量具體結果。

於是交流變成一個 real-time runtime detection 問題：他懂這個詞嗎？他知道我是在類比而不是字面描述嗎？他點頭是真的理解，還是禮貌 ACK？他反對的是結論，還是根本沒有接收到前提？如果沒有相同背景，我又要回去換 encoding，重新發明一種說法。

我目前越來越覺得，一個比較省力的辦法不是「每次都完整解釋」，而是使用分層協議。比如同一個想法可以有幾個版本：

```
level 0: 第一次理解一個東西本來就可能比較慢
level 1: 很多差異其實來自已有模型不同，不一定是能力差異
level 2: 我會把它類比成 cache miss / cache hit
level 3: 進一步討論已有表示、重建成本與遷移
```

先發最小版本，根據對方反應再決定要不要展開。不要一上來就 `level 3 multicast` 全網。🤣

更重要的是，不要只問「懂了嗎？」因為這個問題得到的往往只是：

```
ACK
```

更好的 checksum 是：「你理解成什麼？」或者「我剛才那個比喻在你看來對應的是什麼？」這不是考試，而是在檢查 payload 有沒有 corruption。

5. 真正棘手的不是背景知識不同，而是 EPISTEMIC PROTOCOL 不同

背景知識不同其實還算好處理，最多就是多做幾次 translation。更累的是另一種情況：對方對某些事情有非常根深蒂固的看法，而且幾乎從來不質疑。

這時候雙方的差異不再只是「知道的東西不一樣」，而是「什麼算理由」都不一樣。我可能會問：有什麼證據？有沒有反例？這個結論在哪些假設下成立？有沒有做過實驗？實驗結果如何？從具體結果上升到抽象結論時損失了多少資訊？

對方可能回答：某某教授也這麼說；大家一直都是這樣；我活了這麼多年就是這個經驗；你年紀還小，經驗不足，所以你的想法不靠譜。

這裡需要區分一件事。「某教授也這麼說」並不是天然無效。真正相關的專業判斷本來就可以是 evidence。問題在於，它的權重應該取決於這個人是不是相關領域專家、他的觀點有沒有證據支持、是否存在較穩定的專家共識，而不是：

```
Professor said it -> true
```

同樣，「你經驗不足」最多只能合理降低某些高度依賴 tacit knowledge 的判斷可信度，不能直接推出「你的論證錯」。一個沒開過飛機的人確實很難可靠描述 737 的操縱手感，但如果他指出你的燃油計算少乘了一個係數，回答「你又不是機長」並不能修復那個算術錯誤。💀

更麻煩的是面子。有人即使已經隱約知道自己的想法不完全正確，甚至完全錯了，也可能因為承認錯誤會傷害身份或地位，而拒絕更新。這時候推理會從「找出哪個模型更接近真實」變成「保護既有結論」。人會開始選擇性收集對自己有利的材料，放大支持自己的經驗，淡化反例，用權威替代論證，或者直接攻擊對方的資歷。

這就是 motivated reasoning 最讓人疲憊的地方：推理不再是修改結論的工具，而是保護結論的工具。

「經驗」在這裡尤其容易被濫用，因為經驗本身當然有價值，很多知識甚至只能在長期實踐裡形成；但「我見過很多次」和「這是一條普遍規律」之間仍然隔著很長一段路。經驗可能有 selection bias，可能只覆蓋特定年代、地區或群體，也可能因為記憶會重新編碼而被過度整齊化。最危險的不是使用經驗，而是把經驗當作免檢查的 root certificate：一旦有人說「我活了幾十年」，後面所有 inference 都自動變成 trusted。這在 epistemology 上顯然太奢侈了。🤡

所以我現在更願意把「有經驗」理解成一種 prior 或額外 evidence，而不是最終裁判。經驗可以提高某個判斷的初始可信度，但它仍然可以被更好的資料、更大的樣本、更直接的測量或更完整的機制解釋修正。反過來，純理論也不應該因為形式漂亮就自動戰勝實踐。真正有用的問題仍然是：哪一邊提供了更能解釋現象、也更能承受反例的模型？

6. 比結論更早的一層：我們到底接受了哪些「前提」?

後來我覺得，還有一層比「證據夠不夠」「推理對不對」更容易被忽略：前提。

很多爭論看起來是在爭一個結論，實際上雙方從第一步開始就沒站在同一塊地板上。A 的推理可能完全有效，B 的推理也可能完全有效，但只要兩個人接受的 premises 不同，最後就可以穩定地跑到兩個不同的答案。這時候繼續在結論層互相投擲更多例子，通常只會讓戰線越拉越長。

最簡單的形式大概是：

```
Premise 1
Premise 2
Premise 3
    ↓
Conclusion
```

如果結論錯了，至少有兩種完全不同的可能：一種是 inference 壞了，另一種是 inference 沒問題，但某個 premise 本身不成立。這兩種 bug 不能用同一種方法修。

例如：

```
P1: 有經驗的人通常比沒有經驗的人更可靠
P2: Dorian 在這件事上經驗比較少
C : 所以 Dorian 的觀點不可靠
```

這裡即使暫時接受 P1，結論也仍然跳得太快，因為「通常更可靠」不是「在每一個具體命題上都更可靠」。如果再仔細看，P1 本身也需要限制條件：什麼類型的經驗？和正在討論的命題相關嗎？經驗是否系統性偏斜？是否有更直接的測量可以覆蓋它？所以真正的問題不是一句「尊重經驗」或「不要迷信經驗」就能解，而是要把 premise 拆開。

更麻煩的是，很多前提根本不會被說出來。它們藏在語句下面，作為默認設定運行。比如一個人說：

> 「年輕人就應該先聽過來人的。」

這句話背後可能偷偷掛著很多依賴：

```
older -> more experience
more experience -> better judgement
better judgement in one domain -> better judgement in this domain
past environment ~= current environment
social convention -> normative obligation
```

只要其中一條不成立，整個結論就可能需要重新編譯。💀

這也是為什麼有些對話會讓人有一種很奇怪的疲憊感：你以為自己在反駁一句話，實際上你碰到的是一整棵 hidden dependency tree。你處理完第一層，對方又從下一層推出同一個結論，像永遠清不完的 recursive include。

所以現在如果一個討論長時間卡住，我會開始問：

> 我們是不是其實在不同意前提，而不是不同意結論？

這個問題非常有用，因為它能把很多「你就是不懂」「你太理想化」「你沒有經驗」之類的句子重新拆成可檢查的形式。比如：

「你太理想化」可能真正指的是：

> 我認為某個制度中的參與者會更強烈地追求自身利益，而你給這個因素的權重太低。

這就突然可以討論了。至少我們終於知道 disagreement 在哪。

前提還有另一個很重要的類型：價值前提。有些爭論表面上像是在爭事實，實際上爭的是「我們把什麼當成更重要」。例如：

> 「這個政策是不是好政策？」

這不是只靠一組統計數字就能自動輸出的問題，因為你至少還需要回答：

更重視效率還是公平？
更重視平均結果還是最差者的處境？
更重視短期穩定還是長期自由？
願意接受多大的風險？

兩個人完全可能共享同一批 empirical data，卻因為 value premises 不同而得出不同結論。這時候如果一方一直拿更多數據轟炸另一方，也未必能解決問題，因為分歧不在 data layer。

所以我現在越來越重視一個順序：

```
先找 premises
再看 inference
再看 evidence
最後才看 conclusion
```

而不是一看到結論不同，就直接開始「證明對方錯了」。

這個順序也能避免一個很常見的幻覺：只要推理形式漂亮，結論就可靠。其實形式邏輯只能保證，在前提為真的條件下，某些推理能把真傳到真。它不替你保證前提本身是真的。`valid` 和 `sound` 不是一回事。這一點放到日常討論裡非常重要。

如果要用工程師語言來說：

```
clean inference over garbage premises
= garbage with excellent architecture
```



而且前提的檢查同樣適用於我自己。很多時候我覺得對方「怎麼會得出這種結論」，後來才發現我自己也預載入了幾個沒有顯式聲明的 assumptions，只是因為它們和我的世界觀相容，所以平時看不見。真正麻煩的不是有前提，任何推理都需要前提；真正麻煩的是把自己的前提當成世界的 **default configuration**。

因此，我現在會嘗試把一些關鍵判斷寫成條件句，而不是絕對句：

> 如果 A、B、C 這幾個前提成立，那我目前傾向於 X。

這種說法沒有「X 就是真理」那麼有氣勢，但它多了一個非常重要的接口：一旦 A、B 或 C 改變，我知道應該去哪裡 update，而不是等到整個結論崩掉才開始搶救。

某種意義上，這和數學裡的定理非常接近。數學不會只扔給你一句「因此如此」，它會明確告訴你在什麼 assumptions 下成立。真正乾淨的思考也應該有同樣的習慣：

> 不是只問「你怎麼推出這個結論」，還要問「你允許自己從哪些前提出發」。

所以現在我覺得，在進入長對話之前，有一個問題非常值得先問：

> 什麼證據會讓對方改變看法？

如果答案是「沒有任何證據」，那這場對話就不是 epistemic debate。它可能是身份防衛、地位博弈、價值表態，或者單純是「我不能在你面前認錯」。這時候繼續增加證據，往往只是在提高 data transfer rate，而瓶頸其實在 protocol。

7. 無法證實也無法證偽的問題怎麼辦？

這裡還有一個更麻煩的邊界：有些問題本來就很難通過實驗直接證實或證偽。這時候，如果我只會問「有沒有實驗」，確實會顯得無能為力。

但後來我越來越覺得，這不代表推理失效，只是要換一套工具。經驗命題可以靠觀察和實驗；概念命題要看定義、區分和邏輯一致性；規範命題需要把價值前提說清楚，再看推理是否成立；而一些形上或目前不可判定的命題，最合理的狀態往往不是急著選一邊，而是保留不確定性。

例如：

> 「所有事情都是宇宙特意安排好的。」

如果任何可能發生的結果都可以被重新解釋成「安排的一部分」，那這個命題就沒有給出一個能讓我區分「它為真」和「它為假」的觀察條件。這時候我不需要證明它錯。我只需要說：

> 目前的證據不足以區分這個命題為真或為假。

它的 epistemic status 可以是：

```
underdetermined
```

而不是：

```
false
```

這種克制其實很重要。科學訓練有時候會讓人誤以為所有問題都應該最後輸出 `true` 或 `false`，但很多時候最精確的答案其實是：

```
insufficient information
```

這不是失敗，而是拒絕假裝知道。

而且「不可證偽」也不應該被簡化成一句萬能的「所以它是假的」。更精確的說法是：如果一個命題沒有告訴我們什麼觀察會使它失敗，那我們就很難把它當成一個具有普通經驗內容的科學命題。它仍然可能是某種形上信念、文學性的世界觀、宗教命題或人生態度，但此時討論規則已經改變了。最大的混亂往往不是不同類型的問題共存，而是有人把其中一類的語言搬到另一類裡，卻不承認 protocol 已經切換。

例如，「我選擇相信人生具有某種意義」和「宇宙存在一個可客觀驗證的機制，會根據人的思想頻率安排事件」是兩種完全不同的 claim。前者主要是在表達立場或生活框架，後者則開始對外部世界作出因果描述。一旦走到後者，就不能一邊享受科學語言帶來的權威感，一邊在需要驗證時又退回「這只是你還沒有悟到」。這種雙向逃生艙非常方便，但也正因如此，特別值得警惕。

8. 「別人不懂我」和「我真的說清楚了」是兩回事

我現在回頭看以前那種「別人都不懂我」的感受，會覺得其中有一部分是真的，有一部分其實只是表達沒有完成。

一個想法真正被說清楚，不是我自己覺得「我已經把它說出來了」，而是對方能在自己的腦中重建出一個足夠接近的結構。這意味著我要保的不是原句，不是我最喜歡的比喻，也不是某個漂亮的術語，而是那個 invariant。

比如「cache miss」這個比喻真正想保留的可能只是：

> 第一次處理成本高；已有模型之後成本會下降；成本差異不等於能力差異。

這個結構可以用計算機、走路、學樂器、找文件、做飯來表達。真正高級的表達，不是找到唯一最準確的句子，而是為不同接收者找到最容易讓他重建同一結構的 **encoding**。

從這個角度看，溝通其實很像一個跨模型翻譯問題。你不能假定雙方共享相同 vocabulary、相同 priors、相同 background knowledge，甚至不能假定雙方共享相同的判斷標準。真正麻煩的溝通，經常同時運行：

```
content generation
+ audience modelling
+ error detection
+ translation
+ epistemic negotiation
```

五個 thread 一起跑，風扇當然會起飛。

這也讓我重新理解「會表達」到底是什麼。它不只是詞彙多、句子漂亮，甚至也不只是邏輯正確。真正困難的是，我需要估計對方已經有哪些依賴、缺哪些依賴，然後只補最必要的部分。說得太少，對方無法解碼；說得太多，工作記憶直接 OOM。好的解釋不是把自己知道的全部 dump 出去，而是在盡量少增加額外負擔的情況下，讓對方在自己的認知系統裡重建出相近的模型。

從這個角度看，類比也不再只是修辭裝飾。它其實是一個臨時 adapter：把一個陌生結構映射到對方已有的結構上。但 adapter 也會有 abstraction loss。`cache miss` 可以很好地表達「已有模型會降低後續處理成本」，卻不能自動推出人腦真的按照 CPU cache 的機制工作。如果我忘了這一點，一個本來很好用的類比也會開始偷偷變成理論。這正是我前面批評「頻率」和「能量」時同樣的問題，所以這把刀也必須砍向自己。

9. 不是所有對話都值得跑到收斂

以前我可能更容易默認：既然開始聊了，就應該把事情說清楚，最好最後兩邊能達成某種一致。但現在我越來越覺得，這個目標太昂貴，而且很多時候根本沒有必要。

有些對話的目標只是相處，不需要 epistemic convergence；有些是交換觀點，澄清兩三層就夠了；有些涉及共同決策，那確實值得把模型對齊；有些人只是想表達自己，根本沒有邀請你進行 correction。把所有交流都當成「共同逼近真理」的 session，既不現實，也很耗 CPU。

更重要的是，我現在會試著先判斷：這到底是不是一個求真場景。

如果是，就繼續 debug，檢查定義、證據、反例和推理。如果不是，就不必把「沒能說服對方」誤判成「我的模型失敗」。同時也要保留另一條防火牆：對方拒絕更新，不等於我就一定正確。兩個人完全可以一個很固執，另一個錯得很精緻。

所以我現在更願意採用一個比較穩定的姿勢：

我允許對方不更新，也允許自己錯。

這裡其實還可以再加一個 stop condition。如果一場對話已經反覆出現「重新定義對方的立場」「拒絕給出任何可能改變自己的條件」「把每個反例都事後吸收到原理裡」「從論點轉向身份和資歷」，那麼繼續下去通常已經不再是高收益的求真活動。停止討論不代表同意，也不代表認輸，只代表我不再為一個沒有收斂條件的迭代器提供無限 CPU time。

反過來，如果對方能說「這個證據如果成立，我會把置信度從 80% 降到 50%」，即使最後雙方仍然不同意，我反而會覺得這是一場品質很高的討論。因為真正重要的也許不是同步到同一個答案，而是雙方的模型都保留 update path。

但我希望自己的錯至少是可定位、可修改、可降級置信度的錯，而不是一個靠面子、身份或漂亮詞彙維持的不可觸碰結論。

如果要把這整段經歷壓縮成一句話，大概就是：

> 我不是變得不喜歡抽象了，而是開始要求抽象有 **schema**；不是不相信直覺了，而是開始要求直覺標註 **confidence**；不是一定要把所有人說服，而是先判斷這場對話值不值得支付通信成本。

以前我可能會被一句「理解宇宙的能量，你就會理解人生」吸引一會兒。現在我的第一反應大概是：

```
type error: " 能量" undefined
```

然後補一句：

```
please provide definition, mechanism, evidence, or at least a decent example
```

如果都沒有，那就暫時放回書架。📚😈

10. 我現在真正想保留的，可能不是「理性」，而是可更新性

寫到最後，我發現自己真正珍惜的似乎並不是一種「我要比別人更理性」的身份。那種身份本身也很危險，因為一旦「我是理性的人」變成需要保護的 self-image，我同樣可能開始為自己的結論辯護，只不過把「某某教授說過」換成「我有研究過」「我懂統計」「我讀過 Spivak」。形式更漂亮，但 bug 完全可能一樣。

我更想保留的是 **updatability**。一個觀點可以很強烈，一個模型也可以暫時看起來非常好，但它最好永遠保留幾個接口：什麼證據會讓我調整它？哪些部分是資料支持的，哪些只是我的 extrapolation？哪些概念只是方便的 abstraction，而不是世界本身？如果換一組 assumptions，結論會不會跟著改變？我到底是在描述事實，還是在表達價值？

這也許是數學、寫作和物理最後在我身上匯合的地方。它們表面上教的是完全不同的東西，一個處理證明，一個處理語言，一個處理自然現象，但都在反覆提醒我：不要把沒有展開的直覺冒充完整模型，不要把漂亮的壓縮冒充證明，也不要因為一個想法屬於自己，就免除它被檢查的義務。

真正讓我「回不去」的可能就是這一點。現在看到一個說法，我很難只問「聽起來有沒有道理」，而會自然再多跑幾個 check：

```
What exactly is the claim?
What premises does it depend on?
Which premises are empirical, and which are value judgements?
What do the words mean here?
Does the conclusion actually follow from the premises?
What evidence changes its probability?
What would count against it?
Where did abstraction lose information?
Am I using a metaphor, or claiming a mechanism?
```

這幾個問題有時候確實讓生活變得麻煩一點。很多本來可以愉快接受的「人生智慧」，現在會在本地編譯階段就噴出一屏 warning。🚨 但代價換來的是另一種自由：我不必急著把所有東西判成真或假，也不必因為別人說得很有信心就租用他的 certainty。我可以說「我不知道」「這個部分我只信 60%」「這個比喻很好，但不要往機制上推」「這個問題現在可能沒有足夠資訊」。

如果這也算一種「主角式思考」，那它大概不是相信宇宙會替我安排劇情，而是接受一件平凡得多、也困難得多的事：

> 自己的模型，自己負責維護。 🧠🔧