การวัดผลตัวแทน AI: การสำรวจและแนวโน้มอนาคต

Evaluation of LLM-based Agents: A Comprehensive Survey

การนำทางสู่ยุคใหม่: จากโมเดลภาษาแบบนิ่งสู่ระบบอัตโนมัติที่คิดและทำเองได้

Based on 'Survey on Evaluation of LLM-based Agents' (Yehudai et al., 2025)

NotebookLM

บทนำ: การเปลี่ยนผ่านจากผู้รู้ (LLM) สู่ผู้กระทำ (Agent)

Static LLM (ยุคโมเดลภาษา)

- **การโต้ตอบแบบนิ่ง**: จำกัดอยู่ที่การถาม-ตอบ (Single-turn)
- **Stateless**: ไม่มีการจำค่าสถานะระหว่างขั้นตอน
- **Passive**: รอคำสั่ง ไม่มีการกระทำต่อโลกภายนอก

LLM-based Agent (ยุคตัวแทนอัตโนมัติ)

- **Multi-step Flow**: ทำงานเป็นลำดับขั้นเพื่อแก้ปัญหาซับซ้อน
- **Stateful**: รักษาบริบทและความต่อเนื่องของข้อมูล
- **Environment**: ใช้เครื่องมือ (Tools) และตอบสนองต่อสภาพแวดล้อมจริง

Bottom Line: เกณฑ์วัดผลเดิมไม่เพียงพอ เราต้องการมาตรฐานใหม่เพื่อวัดความสามารถในการ 'กระทำ' (Agency)

NotebookLM

ภาพรวม: 4 มิติสำคัญของการวัดผล

Overview: The 4 Dimensions of Evaluation

1. Agent Capabilities (ความสามารถพื้นฐาน)

Planning, Reasoning, Tool Use, Memory, Self-Reflection

2. Application-Specific (การประยุกต์ใช้เฉพาะทาง)

Web Browsing, Software Engineering, Scientific Research, Conversation

Evaluation Strategy

3. Generalist Agents (ตัวแทน AI อเนกประสงค์)

Operating Systems (OS), Cross-domain tasks, GAIA

4. Frameworks (เฟรมเวิร์กการพัฒนา)

Evaluation throughout the development lifecycle (Dev & Test)

NotebookLM

มิติที่ 1: การวางแผนและการให้เหตุผล (Planning & Reasoning)

Agent ต้องแตกปัญหาใหญ่ให้เป็นงานย่อย และบริหารจัดการลำดับขั้น

The What

ความสามารถในการย่อยปัญหา (Decomposition) และการสร้างเส้นทางสู่คำตอบ (Execution path).

The Benchmarks & Insights

- **Reasoning:** GSM8K, MATH, HotpotQA
- **Planning:** PlanBench, AgentBoard, Natural Plan

Critical Insight:
โมเดลปัจจุบันทำได้ดีในการวางแผนระยะสั้น (Tactical) แต่ยังคงมีปัญหากับงานระยะยาวที่ซับซ้อน (Strategic) ซึ่งต้องการการติดตามสถานะ (State tracking) และการแก้ไขข้อผิดพลาด

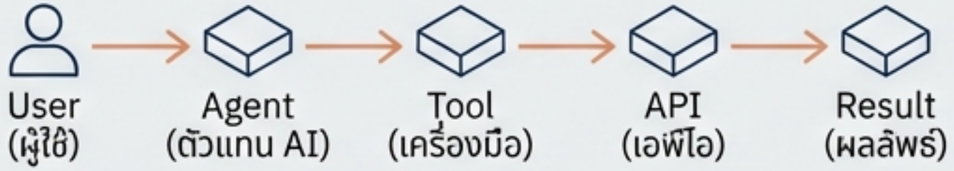
NotebookLM

มิติที่ 1: การใช้เครื่องมือและการสะท้อนคิด
(Tool Use & Reflection)

Function Calling (การเรียกใช้ฟังก์ชัน)

จากการเรียก API ง่ายๆ สู่การจัดการ Tool ที่ซับซ้อนและมีลำดับชั้น (Nested & Stateful).

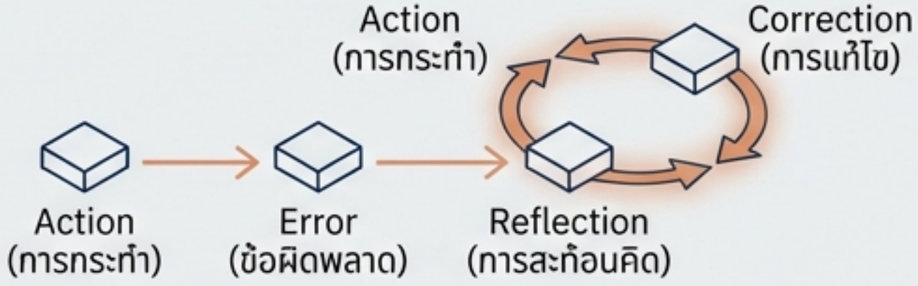
Benchmarks
- **ToolBench**
- **BFCL** (Berkeley Function Calling Leaderboard)
- **API-Bank**



Self-Reflection (การสะท้อนคิด)

ความสามารถในการตรวจสอบและแก้ไขข้อผิดพลาดของตนเอง (Self-correction) เรียนรู้จาก Feedback.

Benchmarks
- **LLF-Bench**
- **Reflection-Bench**



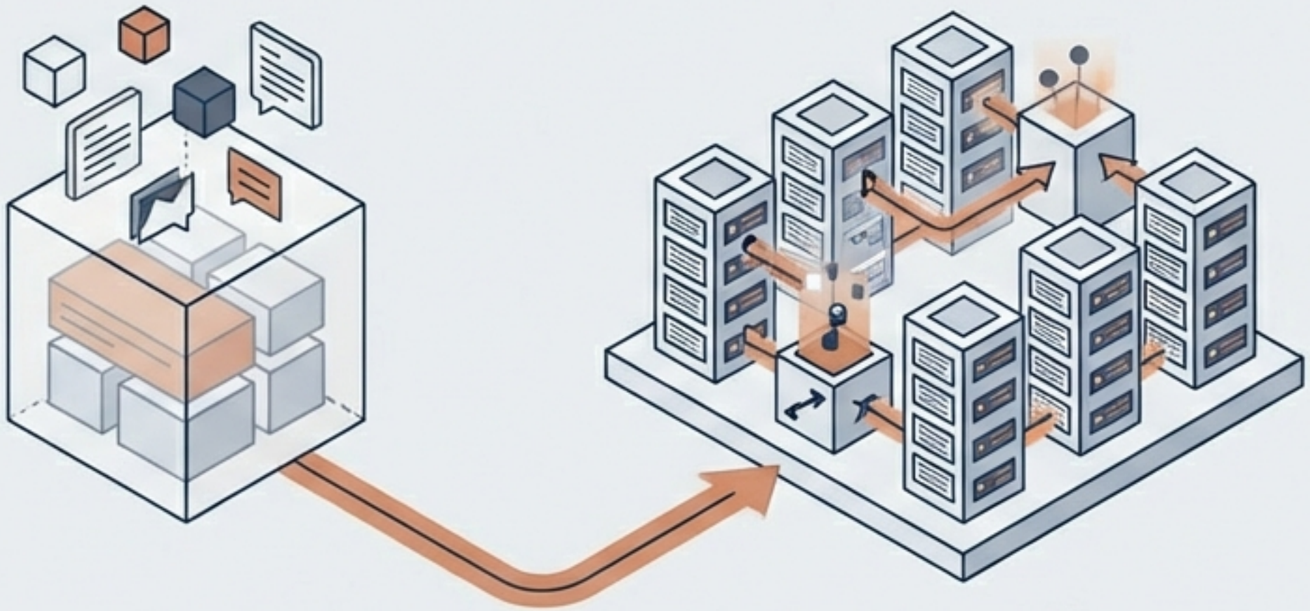
NotebookLM

มิติที่ 1: หน่วยความจำ (Memory)

กุญแจสำคัญสู่ความต่อเนื่องและการเรียนรู้ระยะยาว

Short-term
หน่วยความจำระยะสั้น

Context Window
(หน้าต่างบริบท)



Long-term
หน่วยความจำระยะยาว

Database/Vector Store
(ฐานข้อมูล / เวกเตอร์สโตร์)

Retrieval (การดึงข้อมูล): การดึงข้อมูลจากเอกสารยาว (Benchmarks: *ReadAgent*, *MemGPT*)

Episodic Memory (การจำเหตุการณ์ในอดีต): การจำเหตุการณ์ในอดีตเพื่อปรับปรุงการตัดสินใจในอนาคต

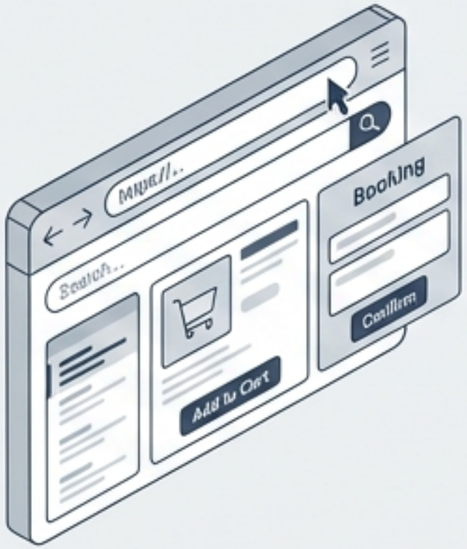
StreamBench
วัดความสามารถในการเรียนรู้ต่อเนื่อง (Continuous improvement) จาก Feedback

MemGPT
จัดการระบบความจำแบบ Tiered memory system (เสมือน OS)

NotebookLM

มิติที่ 2: ตัวแทนสำหรับเว็บและวิศวกรรมซอฟต์แวร์

(Web & Software Agents)



Web Agents (Digital Navigators)

Task: ท่องเว็บ จองตั๋ว ช้อปปิ้ง

Benchmarks

- WebArena
- Mind2Web
- WebLinX

Insight: เปลี่ยนจาก Simulation ง่ายๆ สู่เว็บจริงที่มีความซับซ้อนทางสายตา (Visual Web)

Software Agents (The Coders)

Task: เขียนโค้ด แก้บั๊ก พัฒนาฟีเจอร์

Benchmarks

- SWE-bench (The Gold Standard)
- ใช้ GitHub issues จริง (Real-world repositories)

Insight: ความท้าทายสูงมาก: Success rate ของโมเดลส่วนใหญ่ยังต่ำกว่า 2%

NotebookLM

มิติที่ 2: ตัวแทนด้านวิทยาศาสตร์และการสนทนา
(Scientific & Conversational Agents)

Scientific Agents (Researchers)

Ideation
(ตั้งสมมติฐาน)

→

Experiment Design
(ออกแบบการทดลอง)

→

Code Generation
(เขียนโค้ดวิจัย)

Benchmark: **DiscoveryWorld**, **LAB-Bench**, **SciCode**

เริ่มมีการวัดผล ‘ความคิดสร้างสรรค์’ และคุณภาพการ Peer Review

Conversational Agents (Customer Service)

Focus: การทำตามนโยบายองค์กร (Policy Compliance) และการใช้เครื่องมือระหว่างคุย

Benchmark: **τ-Bench** (Retail/Airline policy), **ABCD**

NotebookLM

มิติที่ 3: ตัวแทน AI อเนกประสงค์ (Generalist Agents)

ความสามารถในการบูรณาการทักษะเพื่อแก้ปัญหาแบบเปิด

Agents ที่ไม่ได้ฝึกมาเพื่องานเดียว แต่สามารถควบคุมคอมพิวเตอร์ได้เหมือนมนุษย์

GAIA

คำถามที่ดูเหมือนง่ายสำหรับมนุษย์ แต่ยากสำหรับ AI เพราะต้องใช้เหตุผล + เครื่องมือ + การท่องเว็บรวมกัน

OSWorld

การควบคุมระบบปฏิบัติการ (Operating System) เต็มรูปแบบ (Multimodal)

AppWorld & OmniACT

การทำงานข้ามแอปพลิเคชันเพื่อทำงานให้สำเร็จ

NotebookLM

มิติที่ 4: ระบบนิเวศสำหรับนักพัฒนา

(The Developer's Ecosystem)

Observability & Evaluation Platforms

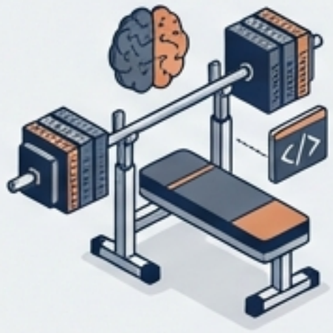
เครื่องมือที่ใช้วัดผลระหว่างการพัฒนาและใช้งานจริง (Development & Deployment)

LangSmith

Langfuse

Galileo

Google Vertex AI



Gym-like Environments

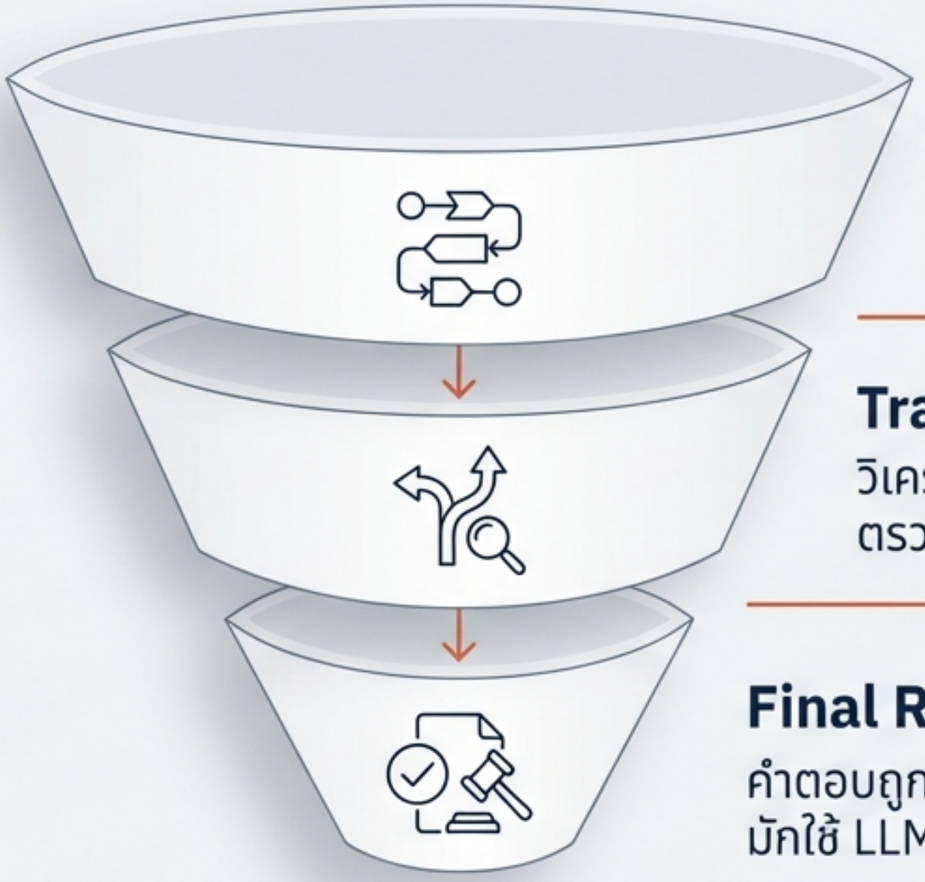
สภาพแวดล้อมจำลองสำหรับการฝึกฝน Agent (Simulated Training)

- **BrowserGym**: สำหรับฝึก Web Agents
- **SWE-Gym**: สำหรับฝึก Coding Agents
- **MLGym**: สำหรับฝึก Research Agents

NotebookLM

วิธีการวัดผลของเฟรมเวิร์ก

(How Frameworks Measure Success)



Stepwise Evaluation (วัดผลรายขั้นตอน)

ตรวจสอบการกระทำทีละขั้น: Agent เลือกเครื่องมือถูกไหม? Parameter ถูกต้องหรือไม่?

Trajectory-Based Assessment (วัดผลเส้นทาง)

วิเคราะห์ลำดับการตัดสินใจเทียบกับเส้นทางที่เหมาะสม (Optimal path). ตรวจสอบว่า Agent วนลูปหรือหลงทางหรือไม่?

Final Response Evaluation (วัดผลลัพธ์สุดท้าย)

คำตอบถูกต้องหรือไม่? (Success/Failure). มักใช้ LLM-as-a-Judge ในการให้คะแนนคุณภาพ

NotebookLM

แนวโน้มปัจจุบัน: ความสมจริงและความท้าทาย

(Current Trends)

Realistic & Challenging (สมจริงและยากขึ้น)

เปลี่ยนจาก Toy problems สู่ Real-world tasks.

Data Point: ใน SWE-bench โมเดลส่วนใหญ่ยังทำคะแนนได้ต่ำมาก สะท้อนความยากของงานจริง

Live Benchmarks (เบนซ์มาร์กที่มีชีวิต)

ข้อมูลหยุดนิ่ง (Static Data) ล้าสมัยเร็วและเสี่ยงต่อการที่โมเดลจำคำตอบ (Data Contamination).

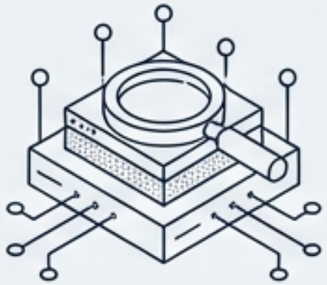
Solution: การอัปเดตข้อมูลต่อเนื่อง เช่น BFCL หรือ SWE-bench Verified ที่มีการปรับปรุง Test cases ตลอดเวลา

NotebookLM

ช่องว่างและทิศทางในอนาคต

(Emergent Directions & Gaps)

Granularity (ความละเอียด)



Problem: รู้แค่ว่า 'ล้มเหลว' แต่ไม่รู้ว่า 'ทำไม'

Need: การวินิจฉัยระดับ Root cause และการวิเคราะห์รายขั้นตอน

Cost & Efficiency (ต้นทุนและประสิทธิภาพ)

Problem: ความเก่งแลกมาด้วยอะไร?

Need: การวัด Token usage, API costs, และเวลาที่ใช้ (Latency)

Safety & Compliance (ความปลอดภัย)

Problem: ยังถูกทดสอบน้อยเกินไป

Need: การป้องกัน Bias, การโจมตี (Adversarial attacks), และการปฏิบัติตามกฎองค์กร

NotebookLM

บทสรุป: กลยุทธ์การวัดผลแบบรอบด้าน

Summary: A Multidimensional Strategy

ประเมินทั้ง 4 มิติ
(Capabilities, Domain, Generalization, Frameworks)

มองให้ลึกกว่า
Success Rate
(Trajectory & Cost)

ใช้ Environment-based Evaluation
และ Continuous Monitoring

นักพัฒนาต้องเปลี่ยนจากการใช้ Static Dataset มาเป็นการทดสอบในสภาพแวดล้อมจริงเพื่อความสำเร็จที่ยั่งยืน

NotebookLM

แผนภาพสรุป Benchmark Landscape

Reasoning	Planning	Tool Use	Web/SWE	Science
GSM8K	PlanBench	ToolBench	WebArena	DiscoveryWorld
MATH	AgentBoard	BFCL	SWE-bench	SciCode
HotpotQA	Natural Plan	API-Bank	Mind2Web	LAB-Bench
StrategyQA	**Science**	NexusRaven	**Generalist**	**Memory**
	DiscoveryWorld		GAIA	MemGPT
	SciCode		OSWorld	StreamBench
			AgentBench	LLF-Bench

รวม Benchmark สำคัญสำหรับการวัดผล Agent ในปัจจุบัน

NotebookLM