

AI 高性能化でハーネスは不要になるのか

— YouTube 動画「【転換点】AI 高性能化でハーネスは不要に？
研究が暴く真実」の一次ソース検証 —

2026 年 10 月 26 日

Claude Opus 5.5

要旨

本書は、YouTube 動画「【転換点】AI 高性能化でハーネスは不要に？研究が暴く真実」^[1]の主張を、動画が引用する論文 3 本と講演 1 本の一次ソースに照らして検証したものである。

結論：動画の骨子——強いモデルでは「知能を補う足場」は効果を失う一方、実行基盤は残る——は一次ソースと整合する。ただし、(1) 数字の文脈の欠落、(2) 枠組みの出典の取り違い、(3) 不利なデータの省略が見られ、「研究が暴く真実」という表現は言い過ぎである。

より正確には、ハーネスの価値は「モデル能力 × タスクの新しさ × 予算」で決まり、モデルが学習済みの領域から順に薄くなる、と整理できる。

1 調査の対象と方法

- **対象：**依頼者から提供された英語の文字起こし（Scripsy.app 作成）^[1]。動画ページには直接アクセスできず、チャンネル名・投稿日は未確認である。
- **方法：**動画中の日付・人名・数値を手掛かりに一次ソース（arXiv 論文本文、講演動画）を特定し、本文で内容を確認した。
- **表記：**【事実】は一次ソースで確認した内容、【分析】は筆者の評価・推論を示す。講演は動画を視聴できておらず、要約記事などの二次情報に依拠するため【事実（二次情報）】と表記する。

2 動画の論旨

動画は、10 月 22 日に「ハーネス設計でスコアが急上昇する」話題を扱った回の続編として、「AI が賢くなるほどハーネスは不要になり、むしろ邪魔になる」という現場の議論を取り上げる^[1]。論旨は次の 4 段で構成される。

- (1) 9月30日の論文を引き、同じ強力なモデルなら、複雑なハーネスは最小構成のエージェントに優位性を示さなかったと紹介する。
- (2) 9月26日の研究を引き、定型的なバグ修正ではハーネスの上乗せ効果が縮むが、リポジトリ全体を扱う自律性の高いタスクでは強いモデルでも効果が残ると留保する。
- (3) 9月17日の論文を根拠に、ハーネスを「知能を補う仕組み」と「安全に運用するための実行基盤」に二分し、前者は自然に消えていくと述べる。
- (4) 元 Cursor のエンジニアの講演を引き、ハーネスは「知能の補助輪」から「未知の領域へ踏み出す乗り物」へ役割を変えたと結ぶ。

最後に、人間の役割は細かな指示から、最小限の実行環境と検証の仕組みの提供へ移る、とまとめている^[1]。

3 照合結果の一覧

表1 動画の引用元と照合結果

動画での言及	特定した一次ソース	判定	主な問題点
9/30 Hernández-Cano チームの論文	Brilliantov ほか (EPFL / Apple) arXiv:2609.40303 ^[2,3]	○ 概ね正確	推論コストの逆転、弱いモデルでの逆転などの留保を省略
9/26 「10 モデル比較」の研究	Hu ほか (南京理工大学ほか) arXiv:2609.32459 ^[4]	△ 一部不正確	7.37pt・6.21pt の比較対象とタスクを誤認
9/17 「4 モデル・176 設定」の論文	Fan ほか (UMass Amherst / Zoom ほか) arXiv:2609.20804 ^[5]	△ 一部不正確	「2 分類」は同論文の結論ではない
「J. Di-A-Cats (元 Cursor、aceXAI)」の講演	Jediah Katz (SpaceXAI) AI Engineer 講演 ^[6,7]	○ 概ね正確	新領域での足し戻し、コストの論点を省略

4 個別検証

4.1 Apple / EPFL 論文 (2026 年 9 月 30 日投稿)

【事実】論文の内容

- 著者は Brilliantov と Hernández-Cano（共同筆頭、Apple でのインターン中の研究）、Abbé で、所属は EPFL と Apple。arXiv への投稿は 9 月 30 日、PDF の日付は 10 月 1 日である [2,3]。
- 自律型の機械学習エンジニアリング（MLE）を題材に、基盤モデル・ハードウェア・時間予算をそろえて比較した。その結果、公開されている最先端の MLE ハーネス 4 種（MLEvolve、AiScientist、Arbor、ScienceFlow）は、read・write・bash を使う単一セッションの最小エージェント「Malena」を上回れなかった。著者らは性能の主因を基盤モデルとし、強いモデルの周りに手作りのハーネスを作り込む労力は、現行のベンチマークでは見合わない結論している [2]。
- GLM 5.2 を使った場合、Malena のメダル獲得率は 62.5%、最良の外部ハーネスは 47.1% であった [2]。

【事実】動画が触れていない点

- **最大の効果は実行環境そのもの：**チャット API でスクリプトを一度に生成させる方式から、ファイル操作とコード実行ができるコーディングエージェント環境へ切り替えたことが、計測した中で最大の効果だった [2]。
- **「最小」でもツールはある：**Malena には、提出の登録、資源と残り時間の確認、バックグラウンドでのジョブ実行のツールが与えられている。24 時間タスクで資源・時間確認ツールとジョブ管理ツールを外すと、メダル獲得率は有意に低下した（ジョブ管理ツールだけを外した場合はわずかに改善） [2]。
- **弱いモデルでは結果が変わる：**小型の Gemma 4 31B では、MLEvolve が平均値で Malena を上回った（信頼区間は重なる） [2]。
- **最小構成の方が推論コストは高い：**GLM 5.2 での推計推論コストは、Malena が 12.12 ドル、AiScientist が 1.95 ドルで、約 6.2 倍である。単一セッションで文脈が膨らみ続け、キャッシュ読み込みが増えるためである。ただし総コストはハードウェア費が支配的とされる [2]。
- **文脈長の制約に弱い：**文脈を 64k トークンに制限すると性能が下がった（信頼区間は重なる） [2]。
- **対象モデルと限界：**使用モデルは GLM 5.2、Kimi K3、DeepSeek V4、Gemma 4 などのオープンウェイトである。著者は限界として、MLE-bench 中心でデータ汚染の可能性があるこ

と、統計的検出力、単一ワーカー内の介入が中心であること、比較対象とのチューニング量の差を挙げている^[2]。

動画の「研究チームは、すべての AI でハーネスが無用だと証明したわけではないと注意している」という説明は、上記の限界記述や、弱いモデルでは手作りのワークフローが依然役立つという論文の記述と整合する^[2]。

【分析】

本論文は「ハーネス不要」ではなく、「シェルとファイルシステムという実行環境（ハーネスの基盤部分）が最も効き、その上に積む探索・協調の仕組みは強いモデルでは効かない」と読むのが正確である。また、最小構成が低コストとは限らない。

4.2 南京理工大学ほかの研究（2026 年 9 月 26 日）

【事実】論文の内容

- 著者は Hu Haichuan ら 8 名（南京理工大学、ミュンヘン工科大学、南京大学）で、日付は 2026 年 9 月 26 日である^[4]。
 - Qwen と DeepSeek の 2 系統からオープンウェイト 10 モデルを選び、最小ハーネスの mini-SWE-agent と製品級ハーネスの OpenCode を、SWE-bench Pro（課題修正）、ProgramBench（リポジトリ生成）、GitTaskBench（リポジトリを活用するタスク）で比較した（計 60 構成）^[4]。
 - 作業手順が比較的固定的な課題修正では、モデル能力が上がるほど複雑なハーネスの上乗せ効果は縮んだ（Qwen の SWE-bench Pro では差が約 6pt から 2.67pt へ）。一方、自由度の高いリポジトリ規模のタスクでは、最新の強いモデルほど複雑なハーネスを活用できた^[4]。
- 動画のこの部分の要約は正確である。

【事実】数字の訂正

動画は「強力な nano-harness が 3.7 モデル比で 7.37 ポイント、DEPV4Pro も 6.21 ポイント改善」と述べるが^[1]、これは誤読である。正しくは次のとおりである^[4]。

- 著者らが 5 つの構成要素（ツール登録、コンテキスト圧縮、明示的計画、サブエージェント、遅延読み込みスキル）を組み込んで自作した軽量モジュール型ハーネス「NanoHarness」を、最小構成の mini-SWE-agent と比べた結果である。

- ProgramBench 全 200 問で、Qwen3.7-Max は 42.88 から 50.25 へ (+7.37pt)、DeepSeek-V4-Pro は 45.14 から 51.35 へ (+6.21pt) 改善した。「3.7」は Qwen3.7-Max、「DEPV4Pro」は DeepSeek-V4-Pro を指す。
- 同条件で製品級の OpenCode と Claude Code は、NanoHarness をさらに 1~2pt 上回った。

【事実】動画が触れていない点

- **逆効果の部品がある**：構成要素別では、ツール登録とタスク特化サブエージェントが安定して効いた。一方、コンテキスト圧縮（Qwen3.7-Max で -4.86pt、DeepSeek-V4-Pro で -3.85pt）と汎用サブエージェントは性能を下げた。リポジトリ生成では詳細な要件やインタフェース制約を長く保持する必要があり、圧縮でそれが失われるためと著者は説明している^[4]。
- **強いモデルでは差がほぼ消える**：SWE-bench Pro の DeepSeek-V4-Pro では、OpenCode が 34.3%、mini-SWE-agent が 33.7%であった^[4]。
- **評価条件と限界**：全モデルを非思考モード・温度 0 で評価している。著者は、構成要素の選択範囲、モデル系統が Qwen と DeepSeek に限られること、主な実験が ProgramBench であることを限界に挙げる^[4]。

【分析】

「タスクの新しさ・自由度によってハーネスの効き方が逆転する」という点は本研究の最も重要な知見であり、動画もこれを正しく伝えている。ただし、部品を選ばなければ逆効果になるという点が抜けている。

4.3 UMass Amherst / Zoom ほかの論文（2026 年 9 月 17 日）

【事実】論文の内容

- 著者は Fan Run-Ze ら 9 名（UMass Amherst、Zoom、Emory 大学、UNC Charlotte）である^[5]。
- 実行ループを固定したまま、計画、行動空間（定義済みツール群か bash のみか）、コンテキスト管理の 3 要素を変え、Nemotron-3（30B・120B・550B）と Mistral-Medium-3.5-128B の 4 モデルについて、SWE-Bench Verified と Terminal-Bench 2.1 で計 176 設定を評価した。「4 モデル・176 設定」は動画のとおりである^[5]。

- **コンテキスト管理**：窓が小さいほど価値が高く、主な効果は窓からあふれて途中終了するのを防ぐことである。管理の有無による差は、SWE-Bench で 32k の 35.7pt から 128k の 2.7pt へ縮む^[5]。
- **計画機能**：弱いモデルでは精度の足場となり、強いモデルでは精度をほぼ変えずにコストを下げる（SWE-Bench で約 30%減）^[5]。
- **行動空間**：定義済みツールは bash が苦手なモデルを助け、bash が得意なモデルは bash のみで低コストになる（Nemotron-3 550B でコスト 53%減）^[5]。

動画による各知見の紹介は正確である。

【事実】動画の誤り：「2 分類」の出典

- 動画は本論文を根拠に「知能を補う仕組み」と「安全に運用するための実行基盤」の 2 分類を示すが^[1]、本論文はそうした分類を結論として示していない。
- 論文は、権限管理（ワークスペース外へのアクセスや破壊的コマンドの拒否など）、編集後の診断、堂々巡りの検知を、全条件で固定した「共通の実行基盤」として扱っている。つまり、それらが必要かどうか自体は検証していない^[5]。
- 「人間が誘導する推論手順の補助は、モデルが賢くなれば自然に消える」という記述も、本論文の主張ではない。

【事実】条件付きの結果

- **bash のみが有利かはタスク次第**：Mistral-Medium-3.5 は、bash のみの環境で Terminal-Bench では成功率が 6.7pt 上がったが、SWE-Bench では 23.2pt 下がった^[5]。
- **計画機能のトレードオフ**：強いモデルでのコスト削減は、成功率のわずかな低下（2.0pt、0.4pt）を伴う^[5]。
- **外部妥当性**：使用モデルは最先端のクローズドモデルではなく、各設定は 1 回のみの実行である。著者は外部妥当性の限界を明記している^[5]。

【分析】

動画の「2 分類」は、後述の Katz 講演や、他の研究（例：Harness-Bench^[8]）に近い整理であり、論文の知見と解説者の解釈が混在している。

4.4 Jediah Katz 講演（AI Engineer、2026 年 10 月）

【事実（二次情報）】講演の内容

- 動画中の「J. Di-A-Cats」「aceXAI」は字幕の誤変換で、正しくは Jediah Katz（SpaceXAI）である。2025 年初めに Cursor に入社し、Cursor のエージェントハーネス担当として最古参のエンジニアであった。SpaceX による Cursor 買収に伴い SpaceXAI へ移った^[7]。
- 講演は AI Engineer での「Is the Agent Harness Dead?」（約 18 分）で、「Smarter Models Killed the Harness. Here's What's Left」の題でも紹介されている^[6,7,9]。
- 主張の骨子は次のとおりである^[7]。高度なハーネスは、もはやコーディング能力を上げない。ハーネスが難しいタスクを可能にし、利用が増えてデータがモデル開発元に渡り、モデルがその技能を内部化すると、ハーネスは足場を削除できる（「データ・フライホイール」）。ただしコーディングがモデルに吸収されると、ハーネスは次の領域を切り開き、循環が再び始まる。
- 6 カ月以内にエージェント作業の 90%がノート PC の外（クラウド）で動くと予測している^[7,9]。

動画の要約は概ね正確である。

【事実（二次情報）】動画が省いた点

- **新領域では足し戻している**：新製品 Grokbot では、モデルがまだ得意でない新しい領域・働き方に対応するため、むしろプロンプトやガードレールを追加している^[7]。
- **モデルが吸収できない 3 要素**：長時間稼働のための実行基盤、信頼とセキュリティ（権限の範囲設定、破滅的な操作や高額請求の防止）、効率（タスクあたりのコスト）の 3 つを挙げている^[7]。動画はファイル操作・権限管理・長期の状態維持には触れているが、コストの論点を落としている。
- **根拠が重複している**：Katz が根拠に挙げたのは、UC Berkeley の研究（HarnessTax^[10]）と、4.1 の Apple / EPFL 論文である^[7]。つまり動画の 4.1 と 4.4 は、独立した裏付けではない。
- **宣伝の側面**：Katz は自社製品の Grokbot を高く評価しており、要約記事も、その部分は独立した知見ではなく提供者の売り込みとして扱うよう注意している^[7]。
- **未解決の緊張**：要約記事は、Katz が同じ研究を「スコア差がない＝足場は不要」と「コスト差がある＝効率が重要」の両方の根拠に使っている点を指摘している^[7]。

留意：講演動画は未視聴であり、本節は AI 生成とみられる要約記事^[7]などの二次情報に依拠している。

5 総合評価【分析】

5.1 妥当な点

- 引用した研究はいずれも 2026 年 9～10 月の最新のもので、選択は適切である。
- 「知能の補助は薄れ、実行基盤と検証は残る」という結論の方向性は、一次ソースと整合する。
- タスクの種類によって結果が変わるという留保を、4.2 の研究に基づいて入れている。

5.2 問題点

- **数字の文脈の欠落：**4.2 の 7.37pt が、何と何をどのタスクで比べた数字かが説明されていない。
- **枠組みの出典の取り違い：**2 分類は 4.3 の論文の結論ではない。
- **不利な証拠の省略：**最小構成の方が推論コストが高い例（4.1）、コンテキスト圧縮が逆効果の例（4.2）、bash のみで 23.2pt 下がった例（4.3）、新領域ではガードレールを足し戻すという講演者自身の発言（4.4）に触れていない。
- **外挿の限界：**3 本の実験はいずれもオープンウェイトのモデルによるコーディング・ML 開発タスクである。最先端のクローズドモデルや、コーディング以外の業務での検証はない。
- **予測と事実の混同：**「推論手順の補助は自然に消える」「6 カ月で 90%が PC の外へ」は予測である。

5.3 より正確な整理

ハーネスの価値は「モデル能力 × タスクの新しさ × 予算」で決まる。モデルが学習済みの領域（定型的なコーディング）から補助輪が外れていき、新しい領域では補助が再び必要になる。実行環境・権限・コスト管理は、モデル能力では代替できない。各研究の知見を区分すると表 2 のとおりである。

表 2 強いモデルで薄れる部分と残る部分

区分	要素	根拠
薄れる部分	木探索・マルチエージェント協調などの探索・協調の作り込み	4.1 ^[2]
薄れる部分	計画機能の精度向上効果（コスト削減効果へ移行）	4.3 ^[5]
薄れる部分	細かく定義したツール群（bash が得意なモデルの場合）	4.3 ^[5]
薄れる部分	コーディング固有の指示や編集補助	4.4 ^[7]
残る部分	シェルとファイルシステムを備えた実行環境	4.1 ^[2]
残る部分	コンテキスト管理（窓が小さい場合）	4.3 ^[5]
残る部分	タスクに特化した構造化ツール・サブエージェント（新しいタスク）	4.2 ^[4]
残る部分	権限・セキュリティ、長時間稼働の基盤、コスト効率	4.4 ^[7]
増える部分	新領域でのプロンプト・ガードレール	4.4 ^[7]

6 知財実務への示唆【分析・推論】

- **知財業務は「新領域」側にある可能性**：特許調査、クレーム作成、拒絶理由への対応は、コーディングほどモデルの学習が進んだ領域ではない。そのため、Katz の言う「新領域ではハーネスを足し戻す」側に近い可能性が高い。
- **圧縮・要約への過度な依存を避ける**：4.2 でコンテキスト圧縮が逆効果だった点は、クレームの構成要件、引用文献の記載箇所、期限など、細部が結論を左右する知財業務で特に重要である。要約に頼らず、原文を参照できる設計が望ましい。
- **補助輪と安全装置を分ける**：モデル更新のたびに、手順指示やテンプレートは一つずつ外して効果を比較する。一方、出典照合、引用文献番号の实在確認、権限・期限管理などの安全装置は維持・強化する。
- **平均点だけで評価しない**：4.1 と 4.4 が示すとおり、スコアが同じでもコストは大きく異なり得る。タスクあたりのコストと、漏れなく仕上がった割合を併せて評価する。

- **小規模な自社検証**：知財タスクでハーネスの効果を直接測った一次研究は今回確認できていない。自社タスク数十件で、最小構成と現行構成を比較する検証が現実的である。

7 未確認事項・留意点

- 対象動画のチャンネル名・投稿日・概要欄は未確認である。内容は提供された文字起こしに依拠しており、字幕の誤変換を含む可能性がある。
- Katz 講演は未視聴であり、AI 生成とみられる要約を含む二次情報^[7,9]に依拠している。
- 引用した論文はいずれも査読前の arXiv プレプリントであり、追試によって結論が変わり得る。
- 4.3 の論文の日付（9 月 17 日）は動画の記載および arXiv 掲載情報に基づく。PDF 本文には日付の記載がない。
- Harness-Bench^[8]と HarnessTax^[10]は前回調査で確認したもので、本書のための再確認は行っていない。

参考文献

（URL はいずれも 2026 年 10 月 26 日閲覧）

- [1] 「【転換点】AI 高性能化でハーネスは不要に？研究が暴く真実」YouTube 動画.
<https://www.youtube.com/watch?v=YKYOtGseblg>（内容は依頼者提供の英語文字起こし
 [Scripsy.app 作成] により確認。動画ページはアクセス制限により直接確認できず、チャンネル
 名・投稿日は未確認）
- [2] K. Brilliantov, A. Hernández-Cano, E. Abbé, "How Much of a Harness Does a Strong Agent Need for Autonomous ML Engineering?", arXiv:2609.40303 (EPFL / Apple, 2026 年 10 月 1 日付 PDF).
<https://arxiv.org/pdf/2609.40303>
- [3] deeplearn.org, 上記[2]の論文掲載情報（投稿日時 2026 年 9 月 30 日の確認に使用、二次情報）.
<https://deeplearn.org/arxiv/836092/how-much-of-a-harness-does-a-strong-agent-need-for-autonomous-ml-engineering>
- [4] H. Hu, Q. Zhang, S. Yu, Z. Chen, T. Luo, C. Fang, Z. Chen, L. Xiao, "Beyond the Model: Demystifying Harness Effects in Software Engineering Agents", arXiv:2609.32459 (南京理工大学・ミ
 ュンヘン工科大学・南京大学, 2026 年 9 月 26 日). <https://arxiv.org/pdf/2609.32459>

- [5] R.-Z. Fan, Z. Zhang, S. Ma, Y. Hu, S. Wang, K. Song, F. Liu, H. Zamani, X. Wang, "An Empirical Study of Harness Design for Coding Agents", arXiv:2609.20804 (UMass Amherst・Zoom・Emory 大学・UNC Charlotte, 2026 年 9 月). <https://arxiv.org/pdf/2609.20804>
- [6] J. Katz, "Is the Agent Harness Dead?" (AI Engineer 講演, YouTube, 2026 年 10 月, 約 18 分) . <https://www.youtube.com/watch?v=7Y4sbZgBdnc> (未視聴)
- [7] BigGo Finance, "Is the Agent Harness Dead? — Jediah Katz, SpaceXAI | AI Engineer" (上記[6]の講演要約, 二次情報) . <https://finance.biggo.com/podcast/119b307cecd92bca>
- [8] Harness-Bench (北京大学ほか) , arXiv:2605.27922 (前回調査で確認。本書では類似の整理の例として言及) . <https://arxiv.org/abs/2605.27922>
- [9] DEV Community, "Sixteen years in, AI writes my code. Here's what I'm still for." (上記[6]の講演への言及を含むブログ, 二次情報) . <https://dev.to/dks/sixteen-years-in-ai-writes-my-code-heres-what-im-still-for-55i4>
- [10] HarnessTax (UC Berkeley / Arena, 2026 年 9 月 16 日) . <https://harnesstax.github.io/> (前回調査で確認)