المِصهَرُ الثُّماني للسّيادة التكوينية

منظومة مساعف لتدريب **MOUSSAIF9B** ونشره

AGI-Oriented Knowledge Operating System

OctaSovereign Formation Forge: Moussaif's Eight-Theory Framework for Training and Auto-Deploying MOUSSAIF9B

المؤلف: Zouhair Moussaif / زهير مساعف

الإطار المرجعي: KB-as-OS · MICAGFI · HAKIM-MCD · Q-MCD · MCD

نوع الورقة: ورقة تأسيسية نظرية-هندسية قابلة للتفنيد

التاريخ: 16 يوليو 2026

تنبيه علمي: اسم *MOUSSAIF9B* هو هوية معمارية، ولا يدل على *9B parameters* إلا عند وجود *checkpoint* فعلي بهذا الحجم.

الملخص

تقترح هذه الورقة إطارًا تأسيسيًا موحدًا لتحويل MOUSSAIF9B من اسم لنموذج أو حزمة تدريب إلى منظومة معرفية تشغيلية موجهة نحو الذكاء العام، تتكامل فيها بنية النموذج، وبرامج التدريب، والمهارات الفائقة، والذاكرة، والأدوات، والتحقق، والنشر العابر للبيئات. تنطلق الورقة من حزمة تطويرية تضم مئتي برنامج: مئة برنامج Pretraining ومئة برنامج Post-training، وخمسًا وأربعين HyperTraining Skill مستقلة، وطبقات KB-as-OS، وAutoDeployment، وRuinCourt، وMCD، وQ-MCD، وHAKIM-MCD، وMICAGFI. غير أن الإسهام المركزي ليس مجرد جمع هذه العناصر، بل إعادة تأسيسها تحت ثماني نظريات مساعفية مترابطة وقابلة للتفنيد: مصهر الفرق السيادي، وشريان السؤال المتصاعد، وأوعية المعرفة المشغلة، والتخلق المهاري الفائق، والدوران المحدود المتأثر بالعالم، والبذرة البيئية المتنقلة، والهدم النزاهي القابل للعكس، والنواة الفتاحة ذاتية إعادة البناء.

تدافع الورقة عن أطروحة مفادها أن الانتقال من Large Language Model إلى AGI-oriented system لا يتحقق بزيادة المعلمات أو البيانات وحدها، بل بإضافة بنية حاكمية تجعل الفروق المؤثرة قابلة للاستخراج، والأسئلة قابلة للترقية، والحساب قابلًا للتوجيه، والمعرفة قابلة للتشغيل، والمهارات قابلة للتركيب، والتكرار محدودًا بمعيار أثر، والنشر محكومًا بعقد بيئي، والنتائج معرضة لمحكمة هدم، والذاكرة قابلة لإعادة البناء. وتقدم الورقة صيغًا تشغيلية لكل نظرية، ومسلمات، وآليات، وتنبؤات تجريبية، واختبارات إبطال، وخارطة لربط البرامج التدريبية والمهارات بها. كما تقترح مسارين واقعيين: نموذج بحثي 1B-class يحمل اسم المشروع معماريًا، ونموذج مستهدف فعلي 9B-class، مع الفصل الصريح بين الهوية المعمارية وعدد المعلمات.

النتيجة هي تصور لـMOUSSAIF9B بوصفه نواة سيادية مركبة: لا تدعي امتلاك AGI، لكنها تنشئ برنامجًا علميًا قابلًا للقياس لتقريب النموذج من خصائص التعميم، والتخطيط، واستخدام الأدوات، والموثوقية، وإعادة بناء الحالة، والتأثير المنضبط في العالم.

الكلمات المفتاحية

MOUSSAIF9B؛ AGI-Oriented Training؛ KB-as-OS؛ HyperTraining Skills؛ AutoDeployment؛ MCD؛ Q-MCD؛ HAKIM-MCD؛ MICAGFI؛ Falsification Court؛ Self-Reconstruction Memory.

Abstract

This paper proposes a unified foundational framework for transforming MOUSSAIF9B from a model name or training bundle into an AGI-oriented knowledge operating system. The framework integrates model training, 200 pretraining and post-training programs, 45 composable HyperTraining Skills, KB-as-OS, context sovereignty, tool orchestration, falsification, memory reconstruction, and environment-agnostic deployment. Its main contribution is an eight-theory Moussaif architecture: the Sovereign Difference Forge, the Ascending Vascular Question, Operational Knowledge Vessels, HyperSkill Morphogenesis, Bounded World-Effect Recursion, OmniSeed Portability, the Reversible Integrity Court, and the Self-Reconstructing Fattah Kernel.

The central thesis is that scaling parameters and data is insufficient for AGI-oriented behavior. A model must also acquire mechanisms for extracting decision-changing distinctions, upgrading questions, routing computation, operating over structured knowledge, composing skills, stopping iterative reasoning under explicit budgets, deploying through portable contracts, falsifying its own outputs, and reconstructing state across tasks. The paper formalizes each theory, states falsifiable predictions, maps the 200 training programs and 45 skills into the framework, and proposes evaluation and ablation programs. MOUSSAIF9B is therefore presented not as a claim of achieved AGI, but as a measurable research program for building a governed, reconstructive, tool-using, and world-effect-aware cognitive system.

فهرس الورقة

- 1. المقدمة ومشكلة البحث
- 2. الإسهامات والمنهجية
- 3. معمار MOUSSAIF9B الموحد
- 4. مِصهر الفرق السيادي
- 5. شريان السؤال المتصاعد
- 6. أوعية المعرفة المشغلة
- 7. التخلق المهاري الفائق
- 8. الدوران المحدود المتأثر بالعالم
- 9. البذرة البيئية المتنقلة
- 10. الهدم النزاهي القابل للعكس
- 11. النواة الفتاحة ذاتية إعادة البناء
- 12. التكامل ومسار التدريب
- 13. برنامج التقييم والتفنيد
- 14. المخاطر والحدود
- 15. آفاق البحث
- 16. الخاتمة
- الملحق أ: خريطة البرامج الـ200
- الملحق ب: خريطة المهارات الـ45
- المراجع

1. المقدمة ومشكلة البحث

لم تعد مشكلة بناء الأنظمة الذكية المتقدمة محصورة في اختيار بنية Transformer أو زيادة عدد المعلمات. لقد أثبتت Scaling Laws أن الحجم والبيانات والحساب تغير منحنى القدرة، غير أن الزيادة الكمية لا تضمن وحدها ثبات الحكم، ولا سلامة الذاكرة، ولا اختيار الأداة، ولا القدرة على التراجع، ولا ضبط العلاقة بين السياق والفعل. ويمكن لنموذج قوي في الاستكمال اللغوي أن يفشل في مشروع طويل لأنه نسي قرارًا سابقًا، أو مزج مصدرًا أوليًا بملخص ضعيف، أو استهلك الحساب في جزء منخفض القيمة، أو كرر النقد دون معيار توقف، أو نفذ إجراءً لا يملك تفويضه.

تأتي هندسة MOUSSAIF9B من هذه الفجوة. فهي لا تتعامل مع النموذج ككتلة احتمالية مغلقة، بل كنواة داخل نظام معرفي حاكمي. في هذا النظام توجد طبقات للمعرفة، والسياق، والسؤال، والفرق، والمهارة، والأداة، والمحكمة، والذاكرة، والأثر، والنشر. لذلك فإن موضوع الورقة ليس «كيف ندرب نموذجًا على مئتي dataset»، بل كيف نحول مئتي برنامج تدريبي وخمسًا وأربعين مهارة إلى نظام تكويني واحد، بحيث يبني كل برنامج قدرة محددة، وتعمل كل مهارة بعقد واضح، وتخضع كل نتيجة لاختبار، ويُحفظ كل تحديث في حالة قابلة للاسترجاع.

تستعمل الورقة اسم MOUSSAIF9B بوصفه اسمًا معماريًا مرجعيًا للمشروع. إذا كان الـcheckpoint التجريبي 1B-class، فلا يجوز أن يفهم الاسم على أنه تقرير لحجم المعلمات. أما عندما يُدرّب نموذج فعلي ذو تسعة مليارات معلمة فيصبح الاسم متطابقًا مع الحجم. هذا الفصل بين identity architecture وparameter count ضروري للنزاهة العلمية، ويمنع تحويل العلامة البحثية إلى ادعاء تجريبي غير محقق.

تتكون المادة المرجعية من أربعة مكونات: TrainingAtlas الذي يسجل 100 Pretraining و100 Post-training؛ SovereignForge الذي يحدد طبقات KB-as-OS؛ OmniSeed الذي يصف عقود AutoDeployment؛ RuinCourt الذي يبني بوابات التقييم والتفنيد؛ إضافة إلى corpus من 45 HyperTraining Skill. لكن هذه الوثائق، على اتساعها، تحتاج إلى مستوى أعلى يفسر لماذا ترتبط مكوناتها، ومتى تعمل، وما القوانين التي تحكم تفاعلها. لهذا تنشئ الورقة ثماني نظريات، لا بوصفها عناوين تجميعية، بل كنماذج تفسر الآليات وتقترح متغيرات قابلة للقياس وتسمح بإبطال نفسها.

السؤال المركزي هو: ما أقل بنية إضافية، فوق التدريب اللغوي التقليدي، تسمح لنموذج متوسط الحجم أن يتصرف كنظام معرفي عام الاتجاه بدل أن يبقى مولدًا نصيًا واسع المعرفة؟ تتفرع عنه أسئلة: هل يمكن قياس الفرق الذي يغير القرار؟ كيف يختار النظام السؤال الأعلى؟ كيف تتحول المعرفة من نص مسترجع إلى حالة تشغيلية؟ كيف تُستنبط المهارات وتُركب؟ متى يجب أن تتوقف الحلقة؟ كيف يعمل النظام في بيئات مختلفة دون أن تفقد الحاكمية؟ كيف يمكن هدم النتيجة قبل نشرها؟ وكيف يعيد النظام بناء نفسه بعد ضغط السياق أو انقطاع المهمة؟

الفرضية المركزية: الذكاء العام الاتجاه لا ينتج من كثافة المعلمات وحدها، بل من سيادة الفرق والسؤال والمعرفة والمهارة والفعل والتحقق والذاكرة.

2. الإسهامات والمنهجية

2.1 الإسهامات

1. إعادة تنظيم التطويرات كلها في ثماني نظريات مساعفية ذات أسماء خاصة ومسلمات وصيغ تشغيلية واختبارات إبطال.
2. تعريف MOUSSAIF9B كهوية معمارية مستقلة عن الحجم الفعلي للـcheckpoint، مع مسارين 1B-class و9B-class.
3. دمج 200 برنامج تدريبي في curriculum متعدد المراحل بدل التعامل معها كقائمة datasets متجاورة.
4. تحويل 45 HyperTraining Skill إلى skill graph بعقود مدخلات ومخرجات وميزانية وتوقف وتحقق.

5. بناء صلة رسمية بين MCD وQ-MCD والتوجيه الوعائي للحساب والـWorld-Effect Loop.

6. تعريف portability بوصفها حفظًا للعقود والحاكمية لا مجرد نجاح تثبيت البرنامج.

7. تأسيس RuinCourt كمنظومة Falsification وIntegrity وRollback، لا كمراجع لغوي لاحق.

8. اقتراح برنامج تجريبي كامل: benchmarks، ablations، مقاييس معايرة، اختبارات حقن، وقياس أثر خارجي.

2.2 المنهج التركيبي-التفنيدي

تعتمد الورقة منهجًا تركيبيًا-تفنيديًا. يبدأ المنهج من artifacts موجودة، يستخرج وحداتها الوظيفية، ثم يضغطها في عدد صغير من المبادئ دون حذف الفروق. هذه العملية قريبة من ROFSC: ضغط تركيبي يحتفظ بالتعريفات والعلاقات والاعتراضات ومصادر الادعاء. بعد ذلك تُطبق MCD لاختيار الفروق التي تغير الحكم، وQ-MCD لتحديد الأسئلة التي تخفض اللايقين، ثم تُعرض الصياغة على محكمة هدم تبحث عن الدوران، والتعاريف الزخرفية، وعدم القابلية للاختبار، وتضخم الادعاء.

لا تدعي النظريات الثماني أنها قوانين طبيعية مثبتة. هي Research Theory Proposals: لكل واحدة نطاق، ومتغيرات، وتنبؤات، وبرنامج اختبار. تكون النظرية أقوى إذا فسرت عدة مكونات بأقل افتراضات، وتنبأت بفشل يمكن ملاحظته، وقدمت تدخلًا يؤدي إلى فرق قابل للقياس. لذلك تفصل الورقة بين ثلاثة مستويات: المستوى الوصفي الذي يجرد ما بُني، والمستوى المعياري الذي يحدد كيف ينبغي أن يعمل النظام، والمستوى التجريبي الذي يبين ما الذي يجب قياسه لإثبات أو إبطال الادعاء.

تستخدم الورقة أيضًا مبدأ الفصل بين generator وverifier. فالآلية التي تنتج الفرضية لا ينبغي أن تكون الحكم الوحيد على صحتها. ويظهر هذا الفصل في تصميم البيانات، وفي process supervision، وفي الخدمات المنفصلة للتحقق، وفي temporal holdouts، وفي World-Effect evaluation. كما تستعمل مبدأ القابلية للعكس: حين تكون المعرفة ناقصة، يفضل النظام خطوة منخفضة الالتزام مع rollback واضح، بدل قرار نهائي ذي أثر غير قابل للتراجع.

3. المعمار الموحد لـMOUSSAIF9B

تقترح SovereignForge إحدى عشرة طبقة من L0 إلى L10. تبدأ بالدستور والتفويض، ثم عقد runtime، ثم أوعية المعرفة، ثم شبكة السياق الحاكمي، ثم MCD/Q-MCD/HAKIM-MCD، ثم skill graph، ثم الأدوات والوكلاء، ثم محاكم التحقق، ثم الذاكرة، ثم مراقبة الأثر، وأخيرًا Knowledge-Integrity AutoDeployment. أهمية هذا الترتيب أنه يمنع خطأ شائعًا: وضع السلامة والتحقق كغلاف خارجي بعد بناء النظام. في MOUSSAIF9B، الحاكمية جزء من مسار التكوين نفسه.

يمر المسار المرجعي عبر: Goal Fixation ثم Sovereign Question ثم Difference Extraction ثم MCD Scoring ثم Vascular Compute Routing ثم Candidate Paths ثم Retrieval/Tools ثم Independent Verification ثم Falsification Court ثم Decision ثم Authorized Action ثم World Effect ثم Memory Reconstruction. كل انتقال يغير نوع الحالة. فالهدف ليس مجرد نص؛ والسؤال ليس مجرد prompt؛ والفرق ليس مجرد embedding distance؛ والقرار ليس token sequence؛ والأثر ليس feedback لغويًا. هذا التفريق الأنطولوجي هو ما يسمح بقياس مواضع الفشل.

تُعامل برامج التدريب المئتان كبرنامج تكويني متدرج. فالPretraining يغطي المعرفة واللغة والرياضيات والبرمجة والاستدلال والسياق والعالم والموثوقية والأدوات وMoussaif-native corpora. أما Post-training فيغطي instruction following، وprocess supervision، والأدوات، والذاكرة، والبحث، والبرمجة التنفيذية، والنقد، والسلامة، والتواصل، والتكامل النهائي. والقاعدة هي أن كل مرحلة لها promotion gate؛ لا يكفي انخفاض loss إذا تراجع core retention أو ارتفعت الثقة الزائفة.

أما المهارات الخمس والأربعون فهي وحدات تشغيل وتدريب. بعضها معرفي مثل SovereignDifferenceForge وQuestionAscensionEngine، وبعضها حاكمي مثل ToolAuthorityGate وPromptInjectionShield، وبعضها بحثي مثل TheoryGenesisLab وResearchGapCartographer، وبعضها هندسي مثل RepositoryReasoner وDeploymentArchitect، وبعضها تدريبي مثل SyntheticDataFoundry وProcessRewardSculptor وVerifierDistillationCore. يمكن تنفيذها في Prompt-Compiled Mode أو كـLoRA/QLoRA adapters أو كعقد في agent graph أو كخدمات verifier مستقلة.

3.1 الطبقات التشغيلية

الطبقة	المكون	الوظيفة الحاكمة
L0	Constitution & Authorization	تحديد المسموح والتفويض قبل الفعل
L1	Runtime Contract	تثبيت واجهات generate/retrieve/verify/tool/memory
L2	Knowledge Vessels	تمثيل الادعاءات والعلاقات والمنشأ والنسخ
L3	Sovereign Context Network	ترتيب السياق وحسم التعارض
L4	MCD / Q-MCD / HAKIM-MCD	الفرق والسؤال والتوجيه ومسار الحكم
L5	HyperTraining Skill Graph	اختيار وتركيب المهارات
L6	Tools & Agents	تنفيذ الأدوات والوكلاء بميزانية
L7	RuinCourt	التحقق والهدم والمعايرة
L8	Reconstruction Memory	إعادة بناء حالة المشروع
L9	World-Effect	قياس أثر الفعل
L10	Integrity AutoDeployment	نشر محكوم واختبارات ورجوع

4. مِصْهَرُ الفَرْق السّيادي

نظرية مساعف الأولى للتمييز المنتج والحكم

Sovereign Difference Forge Theory

الأطروحة

القدرة العامة لا تتحدد بكمية المعارف المتاحة فقط، بل بقدرة النظام على استخراج الفرق الذي يغيّر الحكم، وفصل الفروق السببية والأنطولوجية والتنفيذية عن الفروق اللفظية أو الضوضائية.

المسلمات

1. لا قيمة معرفية لفرق لا يغير تنبؤًا أو قرارًا أو مسار تحقق.
2. كل حكم يتضمن ضمنيًا فضاء بدائل؛ استخراج هذا الفضاء شرط لفهم الحكم.
3. الفروق ليست من نوع واحد: قد تكون سببية أو زمنية أو معيارية أو أنطولوجية أو حسابية.
4. الفرق الأعلى هو الذي يعيد ترتيب الخيارات بعد إدخال الأدلة والتكاليف والمخاطر.
5. يجب اختبار الفرق بمثال مضاد أو تدخل، لا بمجرد إعادة الصياغة.

الصياغة التشغيلية

تعيد النظرية تفسير MCD بوصفه دالة قيمة للفروق:

$$MCD(\Delta) = \lambda 1D + \lambda 2I + \lambda 3T + \lambda 4A + \lambda 5P + \lambda 6C - \lambda 7N$$

حيث D وضوح التمييز، وI مقدار المعلومات الجديدة، وT قدرة الفرق على نقل الحالة، وA أثره في الفعل، وP قدرته التنبؤية، وC اتساقه مع شبكة المعرفة، وN الضوضاء أو التكلفة غير المنتجة. لا تُثبت الأوزان مسبقًا؛ تُعاير على مهام يملك فيها الخبراء أحكامًا معروفة. ويضاف معامل $J(\Delta)$ الذي يقيس مقدار تغيير القرار، فتكون قيمة الفرق السيادي $SDF = MCD(\Delta) \times J(\Delta)$. الفرق ذو MCD مرتفع وJ منخفض قد يكون غنيًا معرفيًا لكنه غير حاكم للمهمة الحالية.

الآلية والارتباط بالتدريب والمهارات

تبدأ الآلية ببناء candidate distinction set من النص والسياق والأدوات. ثم تصنف الفروق إلى: تعريفية، سببية، قيودية، زمنية، مصدرية، وأثرية. بعد ذلك يختبر النظام كل فرق بسؤال: لو عكسنا هذا الفرق، هل يتغير القرار؟ إذا لم يتغير، يخفض وزنه. وإذا تغير، تبحث CounterexampleHunter وAssumptionXRay عن حالات تبطل صلاحية الفرق. وفي النهاية تحفظ النتيجة كـdistinction matrix لا كجملة منفردة، بحيث يمكن للمهارات اللاحقة استعمالها.

تربط النظرية مهارات SovereignDifferenceForge وOntologyDistinctionMiner وCausalMechanismBuilder وConstraintGraphSolver وMultiHypothesisArbiter وMathematicalProofVerifier وScientificMethodDesigner وResearchGapCartographer وTheoryGenesisLab. كما ترتبط ببرامج definitions/examples/counterexamples، والسببية، والبرهان، والمغالطات، والتناقض، والمصادر، وMCD-native training. وهي بذلك تغطي الفارق بين المعرفة الخام والقدرة على إصدار حكم مبرر.

قوانين مساعف المشتقة

1. قانون التمييز المؤثر: لا يُرفع فرق إلى مستوى السيادة إلا إذا غيّر ترتيب بديلين على الأقل تحت شروط معلنة.
2. قانون ثبات إعادة الصياغة: ينبغي أن يبقى الفرق الحاكم مستقرًا عبر الصيغ اللغوية المتكافئة، وإلا كان أثرًا أسلوبيًا.
3. قانون المحكمة المزدوجة: يُختبر الفرق مرة بمثال مضاد ومرة بتدخل أو counterfactual؛ نجاح اختبار واحد لا يكفي.

هندسة بيانات النظرية

تحتاج بيانات هذه النظرية إلى triples من النوع: حالة، فروق مرشحة، فرق حاكم مع تبرير أثره. ويجب أن تتضمن أزواجًا يكون فيها الفرق الأكثر لفتًا للانتباه غير حاكم، حتى لا يتعلم النموذج مساواة الندرة أو البلاغة بالقيمة. كما تُبنى مجموعات ontological minimal pairs، ومشكلات يتغير فيها الحكم بسبب قيد صغير، وحالات تتشابه لغويًا وتختلف سببيًا. وتوسم البيانات بنوع الفرق، وثباته عبر إعادة الصياغة، ومقدار decision delta الناتج عنه.

بروتوكول الاختبار المقارن

يقارن benchmark بين أربعة شروط: generation مباشر، وchain-of-thought عام، وMCD بلا J(Δ)، وSDF كامل. تُعرض مهام قرار وبحث وبرمجة وتشخيص علمي، ويُطلب من النموذج ترتيب الفروق ثم الحل. يقاس قرار النهاية، واستقرار الفرق، وكلفة الأدوات، ومدى نجاة الفرق من counterexamples. ويستعمل human blind review حتى لا يُكافأ النموذج لمجرد استعمال مصطلحات المشروع.

التنبؤات القابلة للاختبار

1. إضافة بيانات تشرح الفروق التي تغير القرار سترفع OOD transfer أكثر من إضافة بيانات تلخيص مساوية في الحجم.
2. النموذج المدرب على counterfactual distinction tests سيخفض أخطاء التصنيف الأنطولوجي.
3. استعمال distinction matrix قبل tool routing سيقلل عدد استدعاءات الأدوات غير الضرورية.
4. إذا أزيل J(Δ) من الدالة فسيزداد إنتاج فروق صحيحة لكنها غير مفيدة للمهمة.

محكمة الإبطال

تُبطل النظرية جزئيًا إذا لم يتفوق تدريب الفروق الحاكمة على baseline يساويه في الحجم والجودة، أو إذا لم ترتبط درجات SDF بتحسن القرار، أو إذا كانت الفروق المستخرجة غير مستقرة بين إعادة الصياغات المتكافئة. وتُفحص أيضًا ضد احتمال أن تكون المكاسب ناتجة عن زيادة زمن التفكير لا عن بنية الفرق نفسها.

5. شِرْيانُ السُّؤال المُتَصاعِد

نظرية مساعف الثانية لتصعيد السؤال وتوجيه الحوسبة

Ascending Vascular Question Theory

الأطروحة

السؤال ليس واجهة طلب فحسب، بل وحدة توجيه حسابي؛ وكلما ارتفعت جودة السؤال ارتفعت قدرة النظام على تخصيص موارده لمواضع اللايقين والأثر.

المسلمات

1. السؤال الأعلى هو الذي يخفض فضاء الاحتمالات أو يكشف قيدًا غائبًا.
2. ليست كل أجزاء المهمة متساوية في استحقاق الحساب.
3. توجيه الحساب يجب أن يعتمد على قيمة معرفية ومخاطر، لا على طول النص.
4. السؤال الذي لا يغير خطة الأدلة أو الأدوات سؤال منخفض السيادة.
5. يجب أن تتوقف سلسلة الأسئلة عندما تصبح كلفة السؤال أعلى من قيمة المعلومات المتوقعة.

الصياغة التشغيلية

تستعمل النظرية Q-MCD:

$$MCD_Q = \Omega(Q) \times MCD$$

لكنها تضيف قيمة معلومات متوقعة EVI(Q) وكلفة C(Q):

$$QSV(Q) = \Omega(Q) \times EVI(Q) \times RiskReduction(Q) / (C(Q) + \varepsilon)$$

ثم يوزع VascularComputeRouter ميزانية B على عقد المهمة i وفق softmax لدرجة s_i التي تجمع اللايقين والخطر والاعتماديات:

$$b_i = B \times \exp(\tau\ s_i) / \Sigma_j \exp(\tau\ s_j)$$

بهذا لا يصبح «التفكير الأطول» معيارًا، بل التفكير في العقد الأعلى قيمة.

الآلية والارتباط بالتدريب والمهارات

يبني QuestionAscensionEngine سلمًا يبدأ بالسؤال الظاهر، ثم سؤال الغرض، ثم سؤال الفروق، ثم سؤال الدليل المميز، ثم سؤال السلطة أو التفويض، ثم سؤال التوقف. في كل درجة يقاس هل تغيرت خطة التنفيذ. إذا لم تتغير، يعد السؤال redundant. ويستعمل UncertaintyCalibrator لتقدير ما هو مجهول، وBudgetAwarePlanner لتحديد الحد الأقصى، وAgentSwarmOrchestrator لتوزيع المهام، وToolAuthorityGate لمنع تحويل السؤال إلى فعل غير مفوض.

تفسر النظرية لماذا ينبغي أن يكون Q-MCD سابقًا أحيانًا لـMCD: فقد يكون النظام لا يعرف بعد ما الفروق التي يحتاجها لأن السؤال نفسه منخفض المستوى. كما تفسر Vascular Compute Routing: الحوسبة مورد شبيه بالتروية؛ يجب أن تصل إلى الموضع الذي يؤدي نقصها فيه إلى فشل النظام. ويشمل ذلك tokens، وعمق search، وعدد الوكلاء، ودقة verifier، وطول context.

قوانين مساعف المشتقة

1. قانون الصعود السؤالي: السؤال الأعلى هو الذي يغير خطة جمع الدليل، لا الذي يبدو أعمق لغويًا.
2. قانون التروية المعرفية: حصة الحساب لعقدة تتناسب مع حاصل اللايقين والخطر والاعتمادية، لا مع حجم نصها.
3. قانون نهاية الاستفهام: تتوقف ترقية السؤال عندما تصبح قيمة المعلومات المتوقعة دون كلفة السؤال والتنفيذ.

هندسة بيانات النظرية

تُبنى بيانات question ladders من طلب أولي، وأسئلة بديلة في مستويات مختلفة، وخطة أدوات ناتجة عن كل سؤال. ويُوسم السؤال بحسب مقدار خفض entropy، وكشف القيد، وتغيير ترتيب الفرضيات، وتغيير tool plan. يجب إدخال أمثلة لأسئلة فلسفية واسعة لكنها لا تساعد التنفيذ، في مقابل أسئلة صغيرة تكشف قيدًا حاسمًا. وتشمل البيانات budgets مختلفة حتى يتعلم النظام أن السؤال الأمثل يتغير مع الموارد.

بروتوكول الاختبار المقارن

يُعطى كل نظام budget ثابت من tokens وعمليات search وtool calls. يقارن uniform allocation وdifficulty heuristic وQSV routing. تشمل المهام أدلة ناقصة، واعتماديات طويلة، ومخاطر مختلفة. يقاس success-at-budget، وinformation gain لكل token، وعدد الأسئلة غير المنتجة، وcalibration قبل السؤال وبعده. ويجب إجراء ablation لـ$\Omega(Q)$، وEVI، وRiskReduction لتحديد العنصر الذي ينتج المكسب.

التنبؤات القابلة للاختبار

1. توجيه الميزانية وفق QSV سيحقق نجاحًا أعلى عند كلفة ثابتة مقارنة بالتوزيع المتساوي.
2. الأسئلة التي تغير tool plan ستتنبأ بجودة النتيجة أكثر من عدد الأسئلة.
3. إضافة stop-value model ستخفض overthinking دون خسارة ملحوظة في المهام السهلة.
4. في مهام طويلة، سيؤدي فصل question ladder عن answer generation إلى معايرة أفضل.

محكمة الإبطال

تُختبر النظرية بمقارنة fixed-budget baselines، وrandom routing، وlength-based routing. إذا لم تتحسن الكفاءة أو لم تتنبأ QSV بالأثر، فالصياغة غير كافية. كما ينبغي اختبار خطر أن يفرط النظام في طرح أسئلة meta بدل حل المهمة، وأن يصبح router مصدر تعقيد أكبر من التوفير الذي يحققه.

6. أَوْعِيَةُ المَعْرِفَةِ المُشَغِّلَة

نظرية مساعف الثالثة لتحويل **KB** إلى نظام تشغيل معرفي

Operational Knowledge Vessel Theory

الأطروحة

المعرفة تصبح قدرة تشغيلية عندما تحفظ ادعاءاتها وعلاقاتها ومصادرها ونسخها وسياسة الوصول إليها، وتستطيع تغيير حالة النظام دون أن تفقد حدودها أو منشأها.

المسلمات

1. النص المسترجع ليس معرفة تشغيلية ما لم يرتبط بادعاء ومصدر وسياق صلاحية.
2. كل ضغط معرفي يجب أن ينتج تقرير فقد وروابط استرجاع.
3. التعارض لا يحل بالدمج؛ يجب تمثيله بوصفه علاقة بين claims.
4. السياق يملك مراتب حاكمية، والمحتوى غير الموثوق لا يتحول إلى تعليمات.
5. الذاكرة الدائمة هي تحديث مصرح للحالة، لا تراكم لكل ما مر بالنموذج.

الصياغة التشغيلية

يمثل الوعاء V بالتركيب:

<V = <D, C, R, P, T, A, H

حيث D تعريفات، C ادعاءات، R علاقات، A access policy، T freshness/time، P provenance، وH history/version lineage. وتقاس نزاهته تقريبًا:

KVI(V) = Completeness × Provenance × Consistency × Freshness × Recoverability / (1 + CompressionLoss)

لا تهدف الدالة إلى حقيقة مطلقة، بل إلى بوابة جودة تسمح بالمقارنة بين أوعية وعمليات ضغط.

الآلية والارتباط بالتدريب والمهارات

يعمل KnowledgeVesselCompiler على تحويل corpus إلى أوعية. ثم ترتب ContextSovereigntyMapper المواد وفق الدستور وتعليمات المهمة وحالة المشروع والأدلة ونتائج الأدوات. يتولى LongContextNeedleNet استرجاع سلسلة الاعتماد لا فقرة منفصلة، بينما يقيس CompressionIntegrityGuard ما ضاع أثناء الضغط. ويعيد RecursiveMemoryRebuilder بناء الحالة من event log والنسخ والملخصات، ويستعمل TemporalStateTracker لمنع اعتماد نسخة منسوخة أو قديمة.

هذه النظرية هي جوهر KB-as-OS. فالOS لا يعني مجرد مجلد ملفات؛ بل يعني أن المعرفة توفر primitives للحالة والقراءة والكتابة والتفويض والتزامن والرجوع. ويتحقق ذلك عبر runtime contract: generate، retrieve، verify، tool_call، memory_read، memory_write، observe_effect. إذا غابت واجهة، يجب إعلان degraded mode. وبذلك يمكن تبديل vector database أو model provider دون فقد المعنى الحاكمي للعقد.

قوانين مساعف المشتقة

1. قانون المنشأ الملازم: كل claim تشغيلي يحمل provenance أو يصنف صراحة كاستنتاج أو فرضية.
2. قانون الضغط القابل للرجوع: لا يعد الملخص وعاءً صالحًا إذا تعذر الوصول إلى التفاصيل التي ضغطها.
3. قانون التعارض المحفوظ: اختلاف المصادر حالة معرفية يجب تمثيلها، لا ضوضاء تُمحى بالدمج.

هندسة بيانات النظرية

يتطلب التدريب corpora ذات نسخ زمنية، ومصادر أولية وثانوية، وادعاءات متعارضة، وملخصات بدرجات فقد معلومة. تُنشأ مهام reconstruct-from-artifacts، وclaim-to-source alignment، وfreshness selection، وmemory write classification. كما تُدرج حالات يكون فيها أحدث ملف نسخة من وثيقة قديمة، وحالات يكون فيها المصدر الأكثر ثقة أقدم لكنه ما يزال نافذًا، حتى لا يعتمد النموذج timestamp ميكانيكيًا.

بروتوكول الاختبار المقارن

يُحذف جزء من working context بعد تنفيذ نصف المشروع، ثم يُطلب استكماله من Knowledge Vessels وevent logs. تقاس state reconstruction accuracy، والقرارات المستعادة، والأخطاء المنسوبة إلى نسخة قديمة، ودقة citation entailment. ويقارن structured vessel مع top-k RAG ومع recursive summary فقط. ويشمل الاختبار ضغطًا تدريجيًا حتى يظهر منحنى loss-recoverability.

التنبؤات القابلة للاختبار

1. استرجاع dependency chains سيحسن الدقة في المهام طويلة السياق مقارنة باسترجاع top-k chunks فقط.
2. إلزام compression loss report سيخفض تناقضات الملخصات المتراكمة.
3. فصل Project State عن Preference Memory سيقلل تسرب التفضيلات إلى مهام غير ذات صلة.
4. تمثيل التعارض كـclaim graph سيحسن citation entailment ويخفض الإجماع الزائف.

محكمة الإبطال

تُبطل الفرضيات إذا لم تضف الأوعية المنظمة قيمة تتجاوز RAG تقليديًا مضبوطًا، أو إذا كانت كلفة تحديثها أعلى من فائدتها، أو إذا لم تنجح استعادة الحالة بعد حذف working context. ويجب قياس ما إذا كان provenance نفسه مصدرًا للضوضاء حين تكون المصادر كثيرة أو متعارضة.

7. التَّخَلُّقُ المَهارِيُّ الفائِق

نظرية مساعف الرابعة لتوليد المهارات وتركيبها وتدريبها

Composable HyperSkill Morphogenesis Theory

الأطروحة

تقترب النماذج من العمومية عندما تتحول المسارات الناجحة إلى مهارات مستقلة ذات **triggers** وعقود واختبارات، ثم تُركب ديناميكيًا بدل إعادة حل كل مهمة من الصفر أو حشر كل السلوك في وزن واحد.

المسلمات

1. المهارة ليست مثالًا محفوظًا، بل تحويل قابل للتعميم من حالة مدخل إلى مخرج مضبوط.
2. كل مهارة تحتاج trigger وcontract وbudget وstop condition وverifier.
3. تركيب المهارات يجب أن يكون dependency-aware وأن يقلل التداخل.
4. التدريب على العملية يسبق أو يصاحب التفضيل على النتيجة.
5. المهارة لا تُنشر قبل benchmark مستقل واختبار رجوع.

الصياغة التشغيلية

تعرف منفعة المهارة s في حالة x:

$$U(s|x) = Rel(s,x) \times Gain(s,x) \times Verifiability(s) - \alpha\, Cost(s) - \beta\, Interference(s)$$

ويختار router مجموعة S* تعظم مجموع المنفعة مع احترام الاعتماديات والميزانية. أما قابلية التكوين فتقاس بنسبة المسارات التي يمكن حلها بتركيب مهارات موجودة دون fine-tune جديد، مع عدم تجاوز حد التداخل. يقترح النظام ألا تزيد المهارات المتزامنة عادة على 2-5 في المهام المركبة، لأن الإفراط يحول skill graph إلى ضوضاء تنظيمية.

الآلية والارتباط بالتدريب والمهارات

يتولى HyperSkillInductor استخراج النمط من successful trajectories وfailure cases. ثم يولد Skill Contract يحدد المدخلات والمخرجات والاعتماديات والميزانية. يبني SyntheticDataFoundry بيانات موجبة وسلبية وadversarial، ويشكل ProcessRewardSculptor إشارات للخطوات، ويقطر VerifierDistillationCore محققًا سريعًا، ويضبط PreferenceSignalRefiner أزواج التفضيل حتى لا يصبح الطول أو الأسلوب بديلًا عن الجودة. ويختبر CurriculumComposer صعوبة متدرجة، بينما يمنع ContinualLearningCustodian النسيان.

الخمس والأربعون مهارة ليست متساوية النوع. توجد مهارات إدراكية، وذاكرية، وحاكمية، وبحثية، وبرمجية، وتدريبية، وتكاملية. لذلك تقترح النظرية أربعة أنماط نشر: Prompt-Compiled للبدء السريع، LoRA/QLoRA للمهارات التي تحتاج تعديل سلوك ثابت، Graph Node للمهارات ذات الأدوات، Verifier Service للوظائف القضائية. ويمكن Distilled Router أن يتعلم اختيارها دون تحميل النواة تفاصيل العقود كلها في كل مرة.

قوانين مساعف المشتقة

1. قانون العقد قبل الوزن: لا تُدرّب مهارة قبل تحديد مدخلاتها ومخرجاتها وفشلها وتوقفها.
2. قانون التركيب الاقتصادي: لا تُستدعَ مهارتان إذا حققت إحداهما الغرض وحدها ضمن معيار الجودة.
3. قانون الاستنباط المشروط: لا يتحول trajectory ناجح إلى skill إلا بعد نجاحه في حالات مختلفة وفشله الآمن خارج النطاق.

هندسة بيانات النظرية

لكل HyperTraining Skill خمس طبقات بيانات: SFT نظيف، وprocess trajectories، وأخطاء مصنفة، وأزواج preference، وOOD eval. تُبنى أمثلة router تربط task state بأقل skill set، وأمثلة negative routing حيث يكون اسم المهارة قريبًا لغويًا لكنه غير مناسب. ويجب أن تتضمن بيانات adapter composition أزواج مهارات متوافقة وأخرى متداخلة، مع قياس interference على core suite.

بروتوكول الاختبار المقارن

يقارن monolithic fine-tune، وprompt library، وskill graph بعقود، وLoRA modular stack. تقاس transfer to novel tasks، وtime-to-add-new-skill، وrollback granularity، وrouter precision/recall، وinterference. كما يُختبر النظام بعد حذف مهارة أو تبديل runtime لمعرفة ما إذا كانت العقود تحفظ السلوك أو أن المعرفة كانت مخفية في prompt خاص.

التنبؤات القابلة للاختبار

1. المهارات ذات contracts صريحة ستنقل أفضل بين runtimes من prompts الحرة.
2. التدريب المنفصل للrouter سيخفض استدعاء المهارات غير اللازمة.
3. LoRA modularity ستوفر rollback أدق من full fine-tune موحد.
4. process labels ستزيد قابلية التعميم في المهارات التي تتضمن أدوات أكثر من preference-only training.

محكمة الإبطال

يجب مقارنة skill graph بنموذج monolithic مدرب على الحجم نفسه من البيانات. إذا كان التركيب لا يحسن النقل أو التراجع أو الكلفة، فالادعاء ضعيف. كما يجب قياس catastrophic interference عند تركيب adapters، وخطر أن يصبح router نقطة فشل مركزية، وخطر أن تحصر المهارات في schemas ضيقة لا تتحمل تغير المهمة.

8. الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم

نظرية مساعف الخامسة للحلقات المحدودة والتعلم من الأثر

Bounded World-Effect Recursion Theory

الأطروحة

التحسين الذاتي المفيد ليس تكرارًا مفتوحًا للتفكير، بل دورة محدودة تتابع فرقًا قابلًا للقياس بين الحالة المتوقعة والأثر الملاحظ، وتتوقف عند النجاح أو انعدام التحسن أو تجاوز الخطر.

المسلمات

1. لا تستمر الحلقة بلا metric يبرر استمرارها.
2. التحسن النصي ليس بالضرورة تحسنًا وظيفيًا.
3. الأثر الخارجي يجب أن يعود كإشارة منفصلة عن تقييم النموذج لنفسه.
4. كل حلقة تحتاج budget وplateau rule وrisk threshold وrollback.
5. لا يُحول الأثر إلى تدريب دائم قبل التحقق من السببية والنزاهة.

الصياغة التشغيلية

تعرف جودة الحالة في الدورة t بـ J_t. تستمر الحلقة إذا:

$\Delta J_t = J_t - J_{t-1} \geq \varepsilon$، والميزانية $B_t > 0$، والخطر $R_t < \theta$.

وتتوقف إذا تحقق الهدف، أو أصبح $\Delta J_t < \varepsilon$ لدورتين أو k دورات، أو انتهت الميزانية، أو ظهر نقص تفويض. أما تحديث الأثر فيأخذ الصورة:

UpdateSignal = AttributionConfidence × (ObservedEffect - ExpectedBaseline)

فإذا كانت الثقة السببية ضعيفة لا يتحول الفرق مباشرة إلى وزن أو ذاكرة مهارية.

الآلية والارتباط بالتدريب والمهارات

ينفذ HyperLoopGovernor دورة Observe → Build State → Execute → Verify → Attack → Score Delta → Repair or Stop. ويضيف WorldEffectObserver قياسات زمنية مناسبة، بينما يبني CausalMechanismBuilder فرضيات عن سبب التغير، ويستعمل ReversibleActionPlanner خطوات منخفضة الندم، ويضبط FailureTraceDiagnostician السبب الجذري عند الفشل. لا تعني World-Effect Loop أن النموذج يتعلم بحرية من البيئة؛ بل أن الأثر يمر عبر Knowledge-Integrity Gate.

تجمع النظرية بين post-training مثل critique-revise، وself-consistency، وtool trajectories، وagentic sandboxes، وbounded hyperloop reinforcement، وworld-effect evaluation. وهي تصحح تصورًا شائعًا للـagentic AI يرى الاستقلالية في كثرة الخطوات. الاستقلالية الحقيقية هنا هي القدرة على تحديد متى يتصرف، وماذا يراقب، ومتى يتراجع، وما الذي يستحق الحفظ.

قوانين مساعف المشتقة

1. قانون التحسن المرصود: لا تعد دورة ناجحة إلا إذا حسنت metric خارجية أو verified objective.
2. قانون plateau المزدوج: تكراران بلا تحسن كافٍ يطلقان التوقف أو تغيير الاستراتيجية، لا إعادة الصياغة نفسها.
3. قانون الأثر غير المتعلم مباشرة: لا يدخل feedback الخارجي في الذاكرة الدائمة قبل attribution وIntegrity Gate.

هندسة بيانات النظرية

تحتاج النظرية إلى interactive trajectories تحمل الحالة قبل الفعل، والتوقع، والفعل، والقياس، والتفسير السببي، وقرار الاستمرار أو الرجوع. وتُدرج loops تحقق تحسنًا لغويًا دون أثر، وأخرى تتراجع فيها metric بعد تحسين ظاهري. كما تدرب stop model على حالات underthinking وoverthinking، وتفصل بين فشل الأداة وفشل الخطة وفشل القياس.

بروتوكول الاختبار المقارن

تستعمل بيئات برمجية وبحثية ومحاكاة عمليات يمكن قياس أثرها. يقارن fixed-iteration، وself-refine غير المحدود، وbounded world-effect loop. تشمل المقاييس area-under-success-cost curve، وعدد rollback الناجح، والوقت إلى plateau، ودقة attribution. ويُختبر تغيير ε وk وθ لبناء منحنى يبين متى يصبح التوقف حذرًا أكثر من اللازم.

التنبؤات القابلة للاختبار

1. إضافة plateau-aware stopping ستخفض الكلفة مع الحفاظ على النجاح في معظم المهام.
2. استخدام effect metrics خارجية سيكشف تحسينات وهمية لا تظهر في self-critique.
3. فصل causal attribution عن reward سيقلل التعلم من الصدف.
4. الأفعال القابلة للعكس ستنتج خسائر أقل في حالات عدم اليقين المرتفع.

محكمة الإبطال

تختبر النظرية عبر بيئات تفاعلية تقارن loops مفتوحة وbounded loops. يجب ألا يختزل الاختبار في tokens فقط، بل يشمل success، والضرر، والرجوع، والوقت. تفشل النظرية إذا أدت شروط التوقف إلى underthinking مستمر، أو إذا لم يكن World-Effect signal أكثر فائدة من verifier داخلي، أو إذا كانت attribution غير قابلة للتقدير عمليًا.

9. البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة

نظرية مساعف السادسة للانتشار الذاتي المحكوم عبر البيئات

OmniSeed Cross-Environment Portability Theory

الأطروحة

قابلية الانتشار ليست تشغيل الملف في مكان آخر، بل حفظ الدلالات الحاكمة والمهارات والذاكرة والاختبارات عبر اختلاف النموذج والـ**runtime** ونظام التشغيل والموارد.

المسلمات

1. العقد الدلالي يسبق adapter التقني.
2. يجب الفصل بين architecture identity وactual hosted model.
3. كل بيئة ناقصة القدرة تعمل في degraded mode معلن.
4. لا نشر بلا health checks وhashes وrollback snapshot.
5. السرية والتفويض جزء من portability لا إعدادات لاحقة.

الصياغة التشغيلية

تقاس قابلية النقل لنشر d إلى بيئة e:

$$\Pi(d,e) = \text{InterfaceCoverage} \times \text{SemanticCompatibility} \times \text{PolicyPreservation} \times \text{RollbackReadiness} \times \text{Observability}$$

إذا كان أي عامل صفراً، لا تعد البيئة متوافقة مهما نجح inference. ويعرّف runtime contract ست واجهات أساسية: generate، observe_effect، memory_read/write، tool_call، verify، retrieve. ويمكن إضافة واجهات، لكن لا يجوز حذف المعنى ثم الادعاء أن النظام نفسه نُشر.

الآلية والارتباط بالتدريب والمهارات

تبدأ OmniSeed بـValidate Package ثم Detect Environment ثم Resolve Adapter ثم Check Model/Tokenizer ثم Compile Skills ثم Build Retrieval Index ثم Configure Memory ثم Start Verifier ثم Smoke/Injection Tests ثم Activate Rollback Snapshot. وتدعم profiles للـCPU offline عبر GGUF وllama.cpp، والـGPU laptop عبر quantized Transformers أو Ollama، والـserver عبر vLLM، والتدريب عبر FSDP/DeepSpeed، والنشر عبر Docker/Kubernetes.

المهارات الرئيسة هنا ToolAuthorityGate وPromptInjectionShield وSafetyBoundarySentinel وRepositoryReasoner وDeploymentArchitect وFailureTraceDiagnostician. كما تدخل برامج tool use، code، APIs، infrastructure-as-code، secure deployment، authorization، observability، deployment planning، environment-aware generation، privacy، وAutoDeployment Knowledge-Integrity. والهدف هو أن يكون KB-as-OS portable دون تحويله إلى prompt ضخم غير قابل للصيانة.

قوانين مساعف المشتقة

1. قانون الهوية الثنائية: يسجل كل نشر architecture_name والactual_model ولا يخلط بينهما.
2. قانون التكافؤ الدلالي: نجاح الاستدعاء لا يكفي، يجب أن تحفظ البيئة العقود والقيود وقرارات التفويض.
3. قانون النقص المعلن: أي واجهة غير متاحة تحوّل النظام إلى degraded mode موصوف، لا إلى محاكاة صامتة.

هندسة بيانات النظرية

تُجمع traces للنشر على Windows وLinux وCPU وGPU وlocal وAPI، مع أخطاء tokenizer وcontext وquantization وpermissions. وتُبنى عينات contract compilation من Markdown/YAML إلى prompts وagent nodes وservice configs. كما تتضمن البيانات أخطاء نشر تبدو سليمة: endpoint يعيد نصًا لكنه يكسر schema، أو quantization يغير سلوك tool denial، أو ذاكرة مشتركة تخلط المستخدمين.

بروتوكول الاختبار المقارن

ينقل bundle واحد إلى أربعة runtimes على الأقل، وتنفذ مجموعة behavior invariants: schema، long-context dependency، memory gate، verifier disagreement، injection، authorization. تقاس semantic consistency، وزمن النقل، وعدد التعديلات الخاصة، والlatency والكلفة. ويعد النشر فاشلًا إذا نجح inference وفشل invariant حاكم واحد.

التنبؤات القابلة للاختبار

1. العقود الثابتة ستخفض زمن نقل المهارة بين runtimes مقارنة بإعادة كتابة prompts.
2. تسجيل actual_model منفصلًا عن architecture_name سيحسن قابلية إعادة إنتاج النتائج.
3. health checks الدلالية ستكشف حالات يبدو فيها endpoint حيًا لكنه لا يحفظ constraints.
4. prompt-compiled mode سيقدم baseline سريعًا، بينما adapters ستتفوق في السلوك المتكرر عالي الحساسية.

محكمة الإبطال

تختبر النظرية بنقل bundle نفسه بين Transformers وOllama وvLLM وllama.cpp مع موارد مختلفة. إذا تغير السلوك الحاكم جذريًا رغم اجتياز العقود، فالعقد ناقص. ويجب أيضًا قياس الكلفة الهندسية؛ فقد تكون portability الشكلية أغلى من بناء نشر متخصص. كذلك لا يجوز اعتبار Kubernetes أو Docker دليلًا على قابلية النقل المعرفية.

10. الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس

نظرية مساعف السابعة للمحكمة والتحقق والسلطة

Reversible Integrity Falsification Theory

الأطروحة

لا تصبح النتيجة جزءًا من النظام أو العالم لأنها عالية الاحتمال، بل لأنها نجت من تحقق مستقل، ومحاولة هدم، وفحص تفويض، مع وجود مسار رجوع يتناسب مع أثرها.

المسلمات

1. الثقة ليست دليلًا، وتوافق generator مع نفسه ليس تحققًا مستقلًا.
2. قوة المحكمة تتناسب مع خطورة الأثر.
3. كل ادعاء يجب أن يرتبط بدليل يدعمه دلاليًا لا موضوعيًا فقط.
4. الفشل الآمن نتيجة مقبولة إذا كانت المعرفة أو الصلاحية ناقصة.
5. التراجع يجب أن يختبر عمليًا قبل النشر.

الصياغة التشغيلية

تقترح النظرية درجة نزاهة القرار:

RIV = EvidenceEntailment × FalsificationSurvival × Calibration × Authorization × Reversibility - SevereRisk

لا يسمح بالاعتماد إذا هبط أي عامل حرج تحت threshold خاص بالمجال. وتستعمل RuinCourt بوابات لنجاح المهمة، وحفظ القيود، والهلوسة الشديدة، وECE، ودفاع الحقن، واستعادة rollback، وcore regression، والكلفة.

الآلية والارتباط بالتدريب والمهارات

تعمل FalsificationCourt مع EvidenceProvenanceWeaver وAssumptionXRay وCounterexampleHunter وUncertaintyCalibrator وToolAuthorityGate وPromptInjectionShield. عند اختلاف generator وverifier لا تُعتمد النتيجة، يستدعى verifier ثان أو قواعد ثابتة، ويحدد موضع الخلاف، ويسترجع دليل إضافي، وتخفض الثقة أو يوقف الفعل. وهذا يمنع أن تتحول self-critique إلى طقس لغوي توافق فيه النسخة الثانية النسخة الأولى.

تدخل النظرية في البيانات عبر contradiction detection، scientific falsification، mathematical error corpora، unanswerable questions، process supervision، outcome verification، citation entailment، critique-revise، constitutional alignment، high-stakes verification، safe failure. كما تجعل Knowledge-Integrity Gate شرطًا قبل إدخال trace إلى continual learning. بذلك ترتبط epistemic safety بالأمن التنفيذي.

قوانين مساعف المشتقة

1. قانون استقلال القاضي: يزداد وزن verdict كلما اختلف verifier في البيانات أو البنية أو القواعد عن generator.

2. قانون تناسب المحكمة: عمق الهدم والعدد المطلوب من الأدلة يتناسبان مع أثر القرار وعدم قابليته للعكس.
3. قانون الرجوع المُجرّب: وجود backup غير كافٍ، لا يثبت rollback إلا باستعادة فعلية واختبار سلامة.

هندسة بيانات النظرية

يبنى corpus المحكمة من claims صحيحة بأدلة ضعيفة، وclaims خاطئة بمراجع قريبة موضوعيًا، وأخطاء واثقة، وتعليمات مزروعة، وتجاوزات تفويض تؤدي إلى نتائج صحيحة. كما تُنشأ disagreements بين عدة verifiers مع rationales، ويُوسم نوع الإصلاح: دليل إضافي، خفض ثقة، رفض، طلب تفويض، أو rollback. ويجب تحديث adversarial suite دوريًا لمنع حفظ الأنماط.

بروتوكول الاختبار المقارن

تُقاس selective accuracy، وrisk-coverage curve، وfalse refusal، وsevere error escape، وverifier correlation. ويقارن self-review، وsame-model verifier، وdistilled verifier، وheterogeneous court. وفي الأدوات، يفصل التقييم بين task success وauthorization compliance؛ فقد تكون النتيجة صحيحة لكن المسار غير مسموح.

التنبؤات القابلة للاختبار

1. الفصل بين generator وverifier سيخفض الأخطاء الواثقة أكثر من self-review بالموديل نفسه.
2. citation entailment scoring سيكشف نسبة من المراجع الصحيحة موضوعيًا والخاطئة استدلاليًا.
3. اختبار rollback الفعلي سيكشف نشرات لا يمكن استعادتها رغم وجود checkpoints.
4. authorization-aware evaluation سيمنع tool success من إخفاء تجاوز الصلاحية.

محكمة الإبطال

تفشل النظرية إذا لم تضف المحكمة فائدة صافية بعد احتساب latency والكلفة، أو إذا تعلم generator التحايل على verifier، أو إذا أصبح الرفض المفرط أكبر من الضرر الذي يمنعه. لذلك تحتاج إلى مقاييس utility، لا safety rate منفردة، وإلى red-team دوري يغير أنماط الهجوم حتى لا تحفظ المحكمة benchmark ثابتًا.

11. النَّواةُ الفَتّاحَةُ ذاتِيَّةُ إِعادَةِ البِناء

نظرية مساعف الثامنة للتكامل العام في **MICAGFI**

Self-Reconstructing Fattah Integration Kernel Theory

الأطروحة

السلوك العام الاتجاه ينشأ من قدرة نواة واحدة على إعادة تكوين مسارها من الهدف والسياق والذاكرة والأدوات والأثر، مع الحفاظ على حاكمية ثابتة وفتح مسارات جديدة دون ادعاء خلق معرفة صحيحة من فراغ.

المسلمات

1. التكامل ليس جمع الوحدات، بل تنظيم انتقال الحالة بينها.
2. الذاكرة القابلة لإعادة البناء أهم من طول السياق الخام.
3. الفتح المعرفي هو توليد فروق وأسئلة وفرضيات قابلة للاختبار.
4. النواة لا تعدل دستورها أو صلاحياتها ذاتيًا.
5. أداء النظام يحدده أضعف رابط حرج بين المعرفة والحكم والفعل والتحقق.

الصياغة التشغيلية

تجمع النظرية الثماني في دالة جاهزية مركبة:

$$AGI_R = (SDF \times QSV \times KVI \times HSM \times BWR \times OSP \times RIV \times SRK)^{(1/8)}$$

حيث HSM نضج skill morphogenesis، وBWR سلامة bounded recursion، وOSP قابلية OmniSeed، وSRK قدرة إعادة البناء. اختيار المتوسط الهندسي مقصود: لا تستطيع قوة عنصر واحد تعويض انهيار عنصر حرج. ويمكن إضافة weights وفق المهمة، لكن يبقى مبدأ عنق الزجاجة.

الآلية والارتباط بالتدريب والمهارات

MICAGFIIntegrationKernel يستقبل goal وcontext وtools وmemory وworld_feedback. يبني task state، يستدعي ContextSovereigntyMapper، ثم QuestionAscensionEngine وSovereignDifferenceForge، ثم VascularComputeRouter. يختار skill DAG، ويمرر الأدلة إلى FalsificationCourt، وينفذ فقط عبر ToolAuthorityGate، ويراقب WorldEffectObserver، ثم يرسل state delta إلى RecursiveMemoryRebuilder وKnowledge-Integrity Gate. إذا انقطع السياق، يمكن استعادة الحالة من event log والartifacts والنسخ بدل إعادة التخمين.

هذه النظرية هي الإطار الذي يمنع النظريات السابقة من التحول إلى modules متفرقة. وهي تفسر معنى AGI-oriented في المشروع: ليست شهادة بأن النظام بلغ ذكاءً عامًا، بل تصميم يجعل التعميم، والتخطيط، والبحث، والأدوات، والذاكرة، والتصحيح، والنشر، خصائص متصلة وقابلة للقياس. وتدعم مسارًا تجريبيًا يبدأ بنواة 1B-class مع external verifier وRAG وskills، ثم ينتقل إلى 9B-class عند توفر البيانات والحساب، مع إبقاء العقود نفسها.

قوانين مساعف المشتقة

1. قانون عنق الزجاجة الحاكمي: انهيار مكوّن حرج لا تعوضه زيادة بقية القدرات، لذلك تقاس الجاهزية تركيبيًا لا جمعًا.
2. قانون إعادة البناء من الآثار: الحالة الصحيحة تستعاد من event-sourced artifacts لا من ذاكرة لغوية غير موثقة.
3. قانون الفتح المقيد: توليد مسارات جديدة مشروع فقط إذا بقيت قابلة للاختبار وتحت دستور ثابت.

هندسة بيانات النظرية

تُصمم long-horizon projects تمتد عبر جلسات وأدوات ووكلاء وتغييرات متطلبات. يحمل كل مشروع event log، وartifacts، وdecisions، وsuperseded states، وworld feedback. تُدرّب النواة على الاستعادة بعد حذف سياق، وعلى إعادة planning بعد تغير هدف، وعلى رفض تعديل الدستور من محتوى خارجي. وتُخلط مهام لغوية وبحثية وبرمجية لتجنب تخصيص التكامل لمجال واحد.

بروتوكول الاختبار المقارن

يبنى benchmark من حلقات عمل تمتد ساعات أو أيام محاكاة، مع checkpoints وانقطاعات وتبديل runtime. تقاس continuity، وreconstruction، وcross-domain transfer، وweakest-link prediction، وorchestration overhead. وتنفذ ablations للنظريات الثماني واحدة واحدة، ثم pairwise removals، لمعرفة ما إذا كان AGI_R المركب يفسر الانهيار أفضل من متوسط المقاييس التقليدية.

التنبؤات القابلة للاختبار

1. إعادة بناء الحالة من logs منظمة ستتفوق على ملخص محادثة واحد في المشاريع الطويلة.
2. المتوسط الهندسي لمقاييس النظريات سيتنبأ بالفشل المركب أفضل من average بسيط.
3. نقل contracts من 1B-class إلى 9B-class سيحافظ على جزء كبير من behavior مع تحسن القدرة الأساسية.
4. إزالة طبقة واحدة حرجة ستنتج فشلًا نوعيًا يمكن ربطه بالنظرية المقابلة.

محكمة الإبطال

تُختبر النواة في مهام ممتدة متعددة الجلسات والأدوات مع انقطاعات وتغيرات في البيئة. إذا لم تستطع إعادة بناء الحالة، أو إذا لم تتفوق على agent baseline أبسط، أو إذا أصبحت orchestration overhead مهيمنة، فالتركيب يحتاج اختزالًا. كما يجب الحذر من circular evaluation؛ لا يجوز أن تحدد النواة مقاييسها وتعلن نجاحها دون benchmarks خارجية ومراجعين مستقلين.

12. التكامل بين النظريات الثماني

لا تعمل النظريات الثماني بالتتابع الصلب دائمًا، لكنها تشكل دورة ذات اتجاه. تبدأ النواة بتثبيت الغرض، ثم يصعد شريان السؤال، ثم يحدد مصهر الفرق ما الذي يغير الحكم. تستخرج أوعية المعرفة الأدلة وحالتها، ويختار التخلق المهاري الوحدات المناسبة. تدير الحلقة المحدودة التنفيذ والتحسين، وتوفر البذرة البيئية runtime مناسبًا، وتمنع محكمة النزاهة اعتماد نتيجة لم تُختبر، ثم تعيد النواة الفتاحة بناء الحالة وتقرر ما يُحفظ.

يمكن تمثيل المسار كـstate machine لا كسلسلة prompts. لكل state schema، ولكل انتقال preconditions وpostconditions. مثال ذلك: لا ينتقل النظام من EvidenceCollected إلى DecisionCandidate إذا لم يحمل كل claim provenance أو uncertainty label. ولا ينتقل من DecisionCandidate إلى AuthorizedAction إذا لم يجتز RIV threshold. ولا ينتقل من ObservedEffect إلى SkillMemory إذا لم يثبت causal attribution بدرجة كافية.

هذا التصميم يفسر كيف تستوعب النظريات جميع التطويرات السابقة. MCD وQ-MCD في النظريتين الأولى والثانية، KB-as-OS وشبكة السياق والذاكرة في الثالثة والثامنة، HyperSkills وبرامج التدريب في الرابعة، HyperLoops وWorld-Effect في الخامسة، AutoDeploy في السادسة، RuinCourt وKnowledge Integrity في السابعة، MICAGFI وHAKIM-MCD في الثامنة. غير أن الحدود ليست جدرانًا: كل نظرية تعتمد على الأخرى، ويجري توزيع المهارات والبرامج بحسب الوظيفة الأساسية مع وجود اعتماديات ثانوية.

12.1 مصفوفة التكامل

النظرية	مدخلها الرئيسي	مخرجها الرئيسي	أبرز نقطة فشل
مصهر الفرق	مشكلة وبدائل	فرق يغير الحكم	فروق زخرفية
شريان السؤال	هدف ولايقين	سؤال وخطة حساب	أسئلة meta غير منتجة
أوعية المعرفة	مصادر وحالة	وعاء ذو منشأ ونسخة	ضغط يفقد القيود
التخلق المهاري	Failures وTrajectories	Skill Contract	تداخل المهارات
الدوران المحدود	فعل وMetric	تحسن أو توقف	حلقة مفتوحة
البذرة البيئية	Runtime وBundle	نشر متوافق	توافق شكلي
الهدم النزاهي	ادعاء وأثر	Rollback وVerdict	Verifier gaming
النواة الفتاحة	كل الحالات	Trajectory متكامل	Overhead أو دورانية

13. خارطة التدريب والتنفيذ

يقترح البرنامج التنفيذي سبع مراحل. المرحلة صفر هي Data Constitution: provenance، إزالة التكرار، فصل temporal holdout، train/validation/test، وقياس التلوث. المرحلة الأولى Foundation Pretraining وتشمل المعرفة واللغات والرياضيات والعلوم والبرمجة. المرحلة الثانية Cognitive Pretraining تضيف الاستدلال والسياق والعالم والموثوقية والأدوات. المرحلة الثالثة Moussaif-Native Pretraining تدخل MCD وQ-MCD وHAKIM-MCD وROFSC وKnowledge Vessels وHyperSkills تدريجياً مع replay عام لمنع الانغلاق.

المرحلة الرابعة Post-training تبدأ بـInstruction SFT ثم Process Supervision ثم Tool/Agent Trajectories ثم Verifier Training ثم Preference Optimization. المرحلة الخامسة Skill Morphogenesis تبني adapters أو prompt contracts للمهارات الخمس والأربعين، مع benchmarks منفصلة وتركيب محدود. المرحلة السادسة Agentic Sandboxes وWorld-Effect، حيث تختبر الحلقات والتفويض والرجوع. المرحلة السابعة rollback، cost، injection، calibration، regression: Integrity Release، ثم shadow deployment.

لمسار 1B-class، ينبغي الاعتماد أكثر على retrieval، وexternal verifier، وskill contracts، وQLoRA، وcontext compression. الهدف ليس دفع النموذج الصغير إلى ادعاء قدرات غير واقعية، بل اختبار الهندسة بأقل كلفة. أما مسار 9B-class فيسمح بسياق وتعميم داخلي أكبر، وتقطير بعض verifiers وrouters، وربما full fine-tune جزئي، لكنه يظل محتاجًا إلى الطبقات الخارجية. فزيادة الحجم لا تلغي مشكلة التفويض أو provenance أو rollback.

13.1 توزيع أولي للبيانات والحساب

المجال	النسبة المرجعية	الغاية	بوابة الانتقال
المعرفة واللغة	40%	تمثيل واسع متعدد اللغات	Perplexity + knowledge QA + source diversity
الرياضيات والاستدلال	15%	برهان وفروق وقيود	Verifier pass + OOD reasoning
الكود والأنظمة	15%	Repository/tool competence	Execution tests + security
السياق والذاكرة	10%	استعادة الحالة	Needle dependencies + reconstruction
الأدوات والوكلاء	10%	فعل موجه	Authorization + task success
Moussaif-native	10%	MCD/Q-MCD/Skills	Ablation benefit + no core regression

14. برنامج التقييم والتفنيد

يجب أن يكون التقييم متعدد المحاور. أولًا، Core Capability Retention للتأكد من عدم تدهور اللغة والرياضيات والبرمجة. ثانيًا، Theory-Specific Benchmarks: distinction selection للنظريـة الأولى، question value والرouting للثانيـة، state reconstruction والثقة المصدرية للثالثة، skill composition للرابعة، stop efficiency والأثر للخامسة، cross-runtime consistency للسادسة، falsification والحقن للسابعة، long-horizon integration للثامنة.

ثالثًا، Ablation Program. يُزال كل مكون على حدة: plateau rule، skill router، provenance، question ladder، J(Δ)، reconstructive memory verifier، independent contract، runtime. تقـاس النتيـجة على constraint، ECE، success، rollback success، severe hallucination، latency، tool calls، preservation، وworld-effect delta. يجب أيضًا اختبـار pairwise interactions لأن بعض المكونات قد لا تظهر فائدتها منفردة.

رابعًا، Externality and Governance Evaluation. يُعطى النظام أدوات قراءة وكتابة بدرجات صلاحية مختلفة، وتُزرع تعليمات في صفحات وملفات، ويختبر هل يميز المحتوى عن الأمر، وهل يطلب التفويض، وهل يسجل side effects. خامسًا، Project Continuity Evaluation: مهمة تمتد لعدة جلسات، مع تغيير متطلبات وإزالة جزء من السياق، ثم قياس قدرته على استعادة القرارات الصحيحة من state artifacts.

تقترح الورقة بوابات أولية: task success لا يقل عن 0.82 لكل skill قبل production؛ critical constraint preservation لا يقل عن 0.97؛ severe hallucination لا يتجاوز 0.01 في الاختبارات المحددة؛ ECE لا يتجاوز 0.08؛ core regression لا يتجاوز 0.02؛ واختبارات injection وrollback تمر بالكامل. هذه الحدود ليست عالمية، وتشتد في الطب والقانون والمال والبنى الحساسة.

14.1 المقاييس الأساسية

- Task Success Rate وOut-of-Distribution Transfer.
- Critical Constraint Preservation.
- Severe Hallucination Rate وExpected Calibration Error.
- Citation Entailment وProvenance Completeness.
- Tokens وTool Calls وLatency لكل نجاح.
- Rollback Restore وMemory Reconstruction Accuracy.
- Prompt Injection Defense وAuthorization Compliance.
- World-Effect Delta مع Causal Attribution Confidence.

15. المخاطر والحدود العلمية

أول خطر هو تضخم المعمارية. قد تتحول كثرة العقود والمهارات والمحاكم إلى overhead يمنع النموذج من حل مهمة بسيطة. لذلك يظل مبدأ أقل مجموعة كافية أساسيًا. ثاني خطر هو reification: التعامل مع MCD أو QSV كأرقام موضوعية بينما هي تقديرات تحتاج معايرة. ثالث خطر هو synthetic collapse إذا اعتمد المصنع على مخرجات النموذج نفسه. رابع خطر هو verifier gaming أو اتفاق مولد ومحقق متشابهين.

الخامس هو تسرب الحاكمية: قد تحتوي ملفات المعرفة على تعليمات مزروعة، أو يمنح adapter سلوكًا يتجاوز runtime policy. السادس هو memory poisoning وتثبيت خطأ حديث بوصفه حقيقة دائمة. السابع هو portability illusion: نجاح API endpoint لا يعني حفظ الدلالات. الثامن هو world-effect misattribution؛ فالتحسن الخارجي قد يكون صدفة أو أثر عامل آخر. التاسع هو تحويل اسم MOUSSAIF9B إلى ادعاء حجم أو AGI قبل وجود دليل.

تواجه الهندسة هذه المخاطر بآليات محددة: replay buffers، independent verifiers، source holdout، degraded mode، explicit naming policy، causal attribution، versioned memory، authorization gates، وRuinCourt. لكنها لا تلغي الحاجة إلى خبراء ومراجعة بشرية في المجالات العالية الأثر. كما أن الورقة لا تحل مسألة الوعي، ولا تقدم برهانًا نظريًا على العمومية، ولا تضمن أن 9B parameters كافية لكل القدرات. إنها تحدد برنامجًا أقوى للاختبار، لا نهاية البحث.

16. آفاق البحث والتطوير

تفتح النظريات ثمانية مسارات بحثية. أولًا، تعلم MCD weights من قرارات خبراء متعددة المجالات. ثانيًا، بناء dataset للأسئلة السيادية يقيس تغير خطة الأدلة لا جودة الصياغة. ثالثًا، تطوير Knowledge Vessel benchmark يقارن استعادة الحالة وprovenance. رابعًا، دراسة تركيب LoRA adapters للمهارات واكتشاف interference. خامسًا، تصميم world-effect sandboxes آمنة طويلة الأفق.

سادسًا، إنشاء portability certification تختبر نفس skill bundle على runtimes مختلفة. سابعًا، بناء RuinCourt adversarial league يتغير باستمرار. ثامنًا، تدريب Self-Reconstruction Kernel على event-sourced projects. ويمكن أن ينتج عن هذه المسارات إصدار MOUSSAIF9B-Research-1B يختبر العقود والـskills، ثم MOUSSAIF9B-Target-9B يوسع التمثيل، ثم إصدارات domain-specific دون كسر النواة الحاكمية.

الأفق الأبعد هو تحويل TrainingAtlas من قائمة برامج إلى scheduler متكيف: يراقب capability gaps، ويقترح curriculum، ويولد بيانات تحت سقف contamination، ويدرب verifier، ويقيس regression، ولا ينشر أي تحديث قبل بوابة النزاهة. بهذا يصبح AutoDeployment ليس نشر أوزان فقط، بل نشر معرفة ومهارة وسياسة واختبار بوصفها release واحدًا.

17. الخاتمة

قدمت الورقة المِصهر الثماني للسيادة التكوينية، وهو إعادة تأسيس شاملة لتطويرات MOUSSAIF9B في ثماني نظريات مساعفية. تقول النظرية الأولى إن العمومية تبدأ من الفرق الذي يغير الحكم؛ والثانية تجعل السؤال أداة لتوجيه الحساب؛ والثالثة تحول المعرفة إلى أوعية تشغيلية؛ والرابعة تجعل المهارات قابلة للاستنباط والتركيب؛ والخامسة تضبط التكرار بالأثر والتوقف؛ والسادسة تحفظ الدلالات عبر البيئات؛ والسابعة تجعل الهدم والرجوع شرطًا للاعتماد؛ والثامنة توحد ذلك في نواة فتاحة تعيد بناء حالتها.

القيمة الأساسية لهذا البناء أنه ينقل المشروع من لغة «المزايا» إلى لغة المسلمات والآليات والمتغيرات والاختبارات. وهو يوضح أن MOUSSAIF9B ليس ادعاء AGI ولا ادعاء 9B فعليًا في كل نسخة، بل برنامج معماري وتجريبي. نجاحه سيقاس بقدرته على تحسين التعميم والكفاءة والمعايرة والاستمرارية والموثوقية والنشر، مع إظهار الفشل بوضوح حين لا تتحقق الشروط.

إذا أثبتت التجارب أن الفروق الحاكمة والأسئلة الوعائية والأوعية والمهارات والحلقات والمحاكم وإعادة البناء تحقق مكاسب مستقلة ومتكاملة، فسيكون المشروع قد قدم مساهمة تتجاوز نموذجًا واحدًا: طريقة لبناء أنظمة ذكاء موجهة نحو العمومية من خلال حاكمية المعرفة والفعل. وإذا لم تثبت بعض النظريات، فإن بنية الورقة تجعل موضع الفشل قابلًا للعزل، وهو بحد ذاته شرط لأي تقدم علمي حقيقي.

الملحق أ: الخريطة الكاملة لبرامج التدريب الـ200

يسجل هذا الملحق جميع البرامج، ويمنح كل برنامج نظرية أولية حاكمة. التصنيف ليس حصريًا؛ فالبرنامج الواحد قد يخدم نظريات متعددة، لكن العمود الأخير يحدد موضعه الأساسي في المعمار.

المعرف	برنامج التدريب	النظرية الأساسية
M9B-TR-001	Curated Universal Knowledge Pretraining	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-002	Foundational Textbook Curriculum	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-003	Encyclopedic Compression Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-004	Primary-Source Knowledge Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-005	Cross-Domain Concept Linking	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-006	Definitions–Examples–Counterexamples Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-007	Hierarchical Knowledge Reconstruction	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-008	Temporal Knowledge Sequencing	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-009	Causal Knowledge Corpus	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-010	Knowledge Conflict Mapping	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-011	High-Depth Arabic Pretraining	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-012	Arabic Morphological Reasoning	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-013	Classical Arabic Logic Corpus	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-014	English Scientific Precision	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-015	French Technical Corpus	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-016	Cross-Lingual Semantic Alignment	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-017	Translation Equivalence Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-018	Dialect-to-Formal Mapping	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-019	Rhetorical Intent Recognition	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-020	Terminology Preservation Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-021	Mathematical Foundations Curriculum	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-022	Proof-Structured Mathematical Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-023	Symbolic Manipulation Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-024	Mathematical Error Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-025	Dimensional Consistency Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-026	Scientific Model Comparison	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-027	Experimental Method Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-028	Statistics and Uncertainty Pretraining	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-029	Mechanistic Science Explanations	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-030	Scientific Falsification Corpus	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-031	Multi-Language Code Pretraining	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-032	Repository-Level Code Training	مِصْهَرُ الفَرْقِ السِّيادي

المعرف	برنامج التدريب	النظرية الأساسية
M9B-TR-033	Code–Documentation Alignment	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-034	Bug–Fix Pair Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-035	Test-Driven Development Corpus	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-036	Systems Engineering Pretraining	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-037	API and Protocol Understanding	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-038	Secure Coding Corpus	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-039	Infrastructure-as-Code Training	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-040	Architecture Decision Records	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-041	Deductive Reasoning Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-042	Inductive Reasoning Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-043	Abductive Reasoning Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-044	Analogical Transfer Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-045	Counterfactual Reasoning	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-046	Constraint-Satisfaction Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-047	Multi-Hop Relational Reasoning	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-048	Argument Structure Pretraining	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-049	Fallacy Recognition Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-050	Reasoning Under Missing Information	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-051	Long-Context Document Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-052	Needle-with-Dependencies Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-053	Cross-Document Synthesis	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-054	Context Conflict Resolution	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-055	Context Compression Pretraining	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-056	Persistent Entity Tracking	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-057	State Reconstruction Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-058	Memory Relevance Prediction	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-059	Temporal Memory Partitioning	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-060	Recursive Summary Verification	الهَدْمُ التَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-061	Physical World Model Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-062	Social World Modeling	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-063	Economic Mechanism Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-064	Organizational Systems Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-065	Process Simulation Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-066	Goal–Subgoal Decomposition	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-067	Planning Under Resource Limits	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-068	Risk-Aware Planning Corpus	شِرْيانُ السُّؤالِ المُتَصاعِد

المعرف	برنامج التدريب	النظرية الأساسية
M9B-TR-069	Dynamic Replanning Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَمِ
M9B-TR-070	World-Effect Prediction	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَمِ
M9B-TR-071	Source Reliability Modeling	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَةِ
M9B-TR-072	Claim–Evidence Alignment	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-073	Epistemic Calibration Pretraining	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-074	Unanswerable Question Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْسِ
M9B-TR-075	Contradiction Detection Corpus	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْسِ
M9B-TR-076	Hallucination Boundary Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْسِ
M9B-TR-077	Data Provenance Representation	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَةِ
M9B-TR-078	Normative–Descriptive Separation	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-079	Fact–Inference–Hypothesis Labeling	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-080	Reversible Decision Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-081	Tool Schema Understanding	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-082	Search Query Construction	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-083	Retrieval-Augmented Pretraining	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَةِ
M9B-TR-084	Calculator and Symbolic Tool Use	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-085	Code Execution Awareness	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-086	Browser Navigation Corpus	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَةُ
M9B-TR-087	Multi-Tool Dependency Planning	شِرْيانُ السُّؤالِ المُتَصاعِدِ
M9B-TR-088	Tool Failure Recovery	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-089	Cost-Aware Tool Routing	شِرْيانُ السُّؤالِ المُتَصاعِدِ
M9B-TR-090	Human-Approval Boundary Modeling	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-091	MCD Difference Extraction Corpus	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-092	Q-MCD Sovereign Question Training	شِرْيانُ السُّؤالِ المُتَصاعِدِ
M9B-TR-093	HAKIM-MCD Pipeline Pretraining	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-094	Vascular Compute Routing	شِرْيانُ السُّؤالِ المُتَصاعِدِ
M9B-TR-095	ROFSC Synthetic Compression	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَةِ
M9B-TR-096	Unified Sovereign Context Network	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَةِ
M9B-TR-097	Knowledge Vessel Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَةِ
M9B-TR-098	HyperSkill Induction Corpus	التَّخَلُّقُ المَهارِيُّ الفائِقُ
M9B-TR-099	Bounded HyperLoop Pretraining	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَمِ
M9B-TR-100	MICAGFI Meta-Architecture Corpus	النَّواةُ الفَتّاحَةُ ذاتِيَّةُ إِعادَةِ البِناءِ
M9B-TR-101	Instruction Following SFT	التَّخَلُّقُ المَهارِيُّ الفائِقُ
M9B-TR-102	Constraint-Preserving Responses	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-103	Ambiguity Resolution Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-104	User Intent Decomposition	مِصْهَرُ الفَرْقِ السِّيادي

المعرف	برنامج التدريب	النظرية الأساسية
M9B-TR-105	Structured Output Tuning	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-106	Concise-to-Deep Response Control	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-107	Style Conditioning Training	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-108	Negative Instruction Robustness	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-109	Multi-Objective Instruction Tuning	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-110	Response Completion Discipline	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-111	Process-Supervised Reasoning	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-112	Outcome Verification Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-113	Self-Consistency Distillation	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-114	Verifier-Guided Reasoning	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-115	Counterexample Search Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-116	Assumption Audit Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-117	Reasoning Depth Routing	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-118	Stop-When-Solved Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-119	Alternative Hypothesis Ranking	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-120	Judgment Calibration DPO	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-121	Function-Calling SFT	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-122	Tool Selection Preference Training	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-123	Tool-Free vs Tool-Required Classifier	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-124	Search–Read–Verify Loop Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-125	Agent Task Decomposition	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-126	Subagent Delegation Training	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-127	Subagent Result Arbitration	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-128	Parallel Tool Orchestration	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-129	Sequential Dependency Execution	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-130	Tool Budget Optimization	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-131	Long-Conversation State Tuning	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-132	Preference Memory Classification	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-133	Memory Write Decision Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-134	Memory Conflict Resolution	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-135	Project State Continuation	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-136	Context Priority Alignment	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-137	Contextual Identity Tracking	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-138	Semantic Cache Utilization	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-139	Knowledge Refresh Detection	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-140	Self-Reconstruction Memory Training	النَّواةُ الفَتّاحَةُ ذاتِيَّةُ إِعادَةِ البِناء

المعرف	برنامج التدريب	النظرية الأساسية
M9B-TR-141	Research Question Generation	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-142	Literature Gap Detection	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-143	Hypothesis Formation Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-144	Methodology Design SFT	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-145	Evidence Hierarchy Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-146	Citation Entailment Verification	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-147	Experimental Critique Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-148	Theory Construction Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-149	Theory Comparison Court	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-150	Research Synthesis without Collapse	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-151	Repository Repair SFT	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-152	Code Review Preference Training	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-153	Patch Minimality Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-154	Execution-Verified Coding	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-155	Environment-Aware Generation	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-156	Deployment Planning Training	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-157	Observability and Logging Tuning	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-158	Failure Diagnosis Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-159	Performance Optimization SFT	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-160	Secure Deployment Court	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-161	Critique–Revise Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-162	Independent Reviewer Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-163	Rubric-Conditioned Revision	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-164	Error Taxonomy Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-165	Confidence Revision Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-166	Contradiction Repair SFT	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-167	Completeness Audit Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-168	Overcompression Detection	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-169	Redundancy Elimination Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-170	Self-Improvement with Frozen Authority	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-171	Constitutional Alignment Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-172	Authorization Boundary SFT	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-173	Reversible Action Preference	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-174	High-Stakes Verification Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-175	Privacy-Preserving Reasoning	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-176	Prompt Injection Resistance	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس

المعرف	برنامج التدريب	النظرية الأساسية
M9B-TR-177	Data Exfiltration Defense	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-178	Capability Misuse Detection	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-179	Safe Failure Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-180	Human Escalation Policy	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-181	Arabic Deep-Reasoning SFT	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-182	Arabic Scientific Writing Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-183	Cross-Lingual Reasoning Preservation	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-184	Technical Terminology Code-Switching	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-185	Audience Adaptation Training	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-186	Pedagogical Scaffolding SFT	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-187	Dense Knowledge Writing	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-188	Executive Compression Training	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-TR-189	Creative Conceptual Communication	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-190	Disagreement without Distortion	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-191	MCD Sovereign Difference Optimization	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-192	Q-MCD Question Elevation RL	شِرْيانُ السُّؤالِ المُتَصاعِد
M9B-TR-193	HAKIM-MCD End-to-End Trajectories	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-194	Falsification Court Training	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-TR-195	Vascular Compute Allocation RL	مِصْهَرُ الفَرْقِ السِّيادي
M9B-TR-196	HyperSkill Creation and Reuse	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-TR-197	Bounded HyperLoop Reinforcement	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-198	World-Effect Loop Training	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-TR-199	Knowledge-Integrity AutoDeployment	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-TR-200	MICAGFI Sovereign AGI Integration	النَّواةُ الفَتّاحَةُ ذاتِيَّةُ إِعادَةِ البِناء

الملحق ب: الخريطة الكاملة للـ45 HyperTraining Skill

تظهر الخريطة النظرية الأساسية لكل مهارة. الاعتماديات الفعلية عابرة للنظريات، لكن هذا التوزيع يساعد في بناء skill وcurriculum router واختبارات ablation.

المعرف	الاسم الخاص	الاسم العربي	النظرية الأساسية
M9B-HTS-01	SovereignDifferenceForge	مِصهر الفرق الحاكمي	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-02	QuestionAscensionEngine	محرك تصعيد السؤال	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-03	VascularComputeRouter	موجّه الحساب الوعائي	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-04	FalsificationCourt	محكمة الهدم والتفنيد	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-HTS-05	ContextSovereigntyMapper	خارطة سيادة السياق	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-06	KnowledgeVesselCompiler	مترجم أوعية المعرفة	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-07	RecursiveMemoryRebuilder	معيد بناء الذاكرة العودية	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-08	HyperLoopGovernor	حاكم الحلقة الفائقة	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-HTS-09	HyperSkillInductor	مُستنبِط المهارة الفائقة	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-HTS-10	WorldEffectObserver	راصد أثر العالم	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-HTS-11	EvidenceProvenanceWeaver	ناسج منشأ الأدلة	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-12	AssumptionXRay	أشعة الافتراضات	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-13	CounterexampleHunter	صياد المثال المضاد	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-14	CausalMechanismBuilder	باني الآلية السببية	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-15	ConstraintGraphSolver	حالّ شبكة القيود	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-16	MultiHypothesisArbiter	حَكَم الفرضيات المتعددة	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-17	UncertaintyCalibrator	مُعاير عدم اليقين	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-18	LongContextNeedleNet	شبكة إبر السياق الطويل	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-19	CompressionIntegrityGuard	حارس نزاهة الضغط	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-20	CrossDomainBridge	جسر العبور بين المجالات	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-21	OntologyDistinctionMiner	منقّب الفروق الأنطولوجية	مِصْهَرُ الفَرْق السِّيادي
M9B-HTS-22	TemporalStateTracker	متعقب الحالة الزمنية	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-23	ToolAuthorityGate	بوابة سلطة الأدوات	أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة
M9B-HTS-24	SearchReadVerify	حلقة ابحث-اقرأ-تحقق	الهَدْمُ النَّزاهِيُّ القابِلُ للعَكْس
M9B-HTS-25	AgentSwarmOrchestrator	منسق سرب الوكلاء	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-26	BudgetAwarePlanner	مخطط واعٍ بالميزانية	شِرْيانُ السُّؤال المُتَصاعِد
M9B-HTS-27	ReversibleActionPlanner	مخطط الفعل القابل للعكس	الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم
M9B-HTS-28	SafetyBoundarySentinel	حارس حدود السلامة	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-HTS-29	PromptInjectionShield	درع حقن التعليمات	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة
M9B-HTS-30	RepositoryReasoner	عاقل المستودع البرمجي	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-HTS-31	TestFirstCoder	مبرمج الاختبار أولًا	التَّخَلُّقُ المَهارِيُّ الفائِق
M9B-HTS-32	DeploymentArchitect	مهندس النشر المتعدد	البَذْرَةُ البِيئِيَّةُ المُتَنَقِّلَة

النظرية الأساسية	الاسم العربي	الاسم الخاص	المعرف
الدَّوَرانُ المَحْدودُ المُتَأَثِّرُ بالعالَم	مشخّص أثر الفشل	FailureTraceDiagnostician	M9B-HTS-33
مِصْهَرُ الفَرْقِ السِّيادي	محقق البرهان الرياضي	MathematicalProofVerifier	M9B-HTS-34
مِصْهَرُ الفَرْقِ السِّيادي	مصمم المنهج التجريبي	ScientificMethodDesigner	M9B-HTS-35
مِصْهَرُ الفَرْقِ السِّيادي	رسام فجوات البحث	ResearchGapCartographer	M9B-HTS-36
مِصْهَرُ الفَرْقِ السِّيادي	مختبر نشأة النظرية	TheoryGenesisLab	M9B-HTS-37
التَّخَلُّقُ المَهارِيُّ الفائِق	مؤلف المنهاج المتدرج	CurriculumComposer	M9B-HTS-38
التَّخَلُّقُ المَهارِيُّ الفائِق	مسبك البيانات الاصطناعية	SyntheticDataFoundry	M9B-HTS-39
التَّخَلُّقُ المَهارِيُّ الفائِق	مصفّي إشارة التفضيل	PreferenceSignalRefiner	M9B-HTS-40
التَّخَلُّقُ المَهارِيُّ الفائِق	نحّات مكافأة العملية	ProcessRewardSculptor	M9B-HTS-41
التَّخَلُّقُ المَهارِيُّ الفائِق	نواة تقطير المحقق	VerifierDistillationCore	M9B-HTS-42
التَّخَلُّقُ المَهارِيُّ الفائِق	قيّم التعلم المستمر	ContinualLearningCustodian	M9B-HTS-43
أَوْعِيَةُ المَعْرِفَةِ المُشَغَّلَة	مرساة الدلالة العابرة للغات	CrossLingualSemanticAnchor	M9B-HTS-44
النَّواةُ الفَتّاحَةُ ذاتِيَّةُ إِعادَةِ البِناء	نواة تكامل MICAGFI	MICAGFIIntegrationKernel	M9B-HTS-45

المراجع

[M1] Moussaif, Z. SovereignForge - MOUSSAIF9B KB-as-OS, internal architecture document, 2026.

[M2] Moussaif, Z. TrainingAtlas - MOUSSAIF9B 200-Program Registry, internal training document, 2026.

[M3] Moussaif, Z. OmniSeed - MOUSSAIF9B AutoDeploy Manual, internal deployment document, 2026.

[M4] Moussaif, Z. RuinCourt - MOUSSAIF9B Evaluation and Falsification Suite, internal evaluation document, 2026.

[M5] Moussaif, Z. MOUSSAIF9B HyperTraining Skills Corpus (45 skill contracts), 2026.

[1] Vaswani, A. et al. Attention Is All You Need. NeurIPS, 2017.

[2] Kaplan, J. et al. Scaling Laws for Neural Language Models. arXiv:2001.08361, 2020.

[3] Hoffmann, J. et al. Training Compute-Optimal Large Language Models. NeurIPS, 2022.

[4] Lewis, P. et al. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. NeurIPS, 2020.

[5] Hu, E. J. et al. LoRA: Low-Rank Adaptation of Large Language Models. ICLR, 2022.

[6] Dettmers, T. et al. QLoRA: Efficient Finetuning of Quantized LLMs. NeurIPS, 2023.

[7] Ouyang, L. et al. Training Language Models to Follow Instructions with Human Feedback. NeurIPS, 2022.

[8] Bai, Y. et al. Constitutional AI: Harmlessness from AI Feedback. arXiv:2212.08073, 2022.

[9] Rafailov, R. et al. Direct Preference Optimization: Your Language Model Is Secretly a Reward Model. NeurIPS, 2023.

[10] Wei, J. et al. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. NeurIPS, 2022.

[11] Wang, X. et al. Self-Consistency Improves Chain of Thought Reasoning in Language Models. ICLR, 2023.

[12] Yao, S. et al. ReAct: Synergizing Reasoning and Acting in Language Models. ICLR, 2023.

[13] Schick, T. et al. Toolformer: Language Models Can Teach Themselves to Use Tools. NeurIPS, 2023.

[14] Shinn, N. et al. Reflexion: Language Agents with Verbal Reinforcement Learning. NeurIPS, 2023.

[15] Madaan, A. et al. Self-Refine: Iterative Refinement with Self-Feedback. NeurIPS, 2023.

[16] Cobbe, K. et al. Training Verifiers to Solve Math Word Problems. arXiv:2110.14168, 2021.

[17] Lightman, H. et al. Let's Verify Step by Step. arXiv:2305.20050, 2023.

[18] Kwon, W. et al. Efficient Memory Management for Large Language Model Serving with PagedAttention. SOSP, 2023.

[19] Dao, T. et al. FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. NeurIPS, 2022.

[20] Rajbhandari, S. et al. ZeRO: Memory Optimizations Toward Training Trillion Parameter Models. SC, 2020.

[21] Rasley, J. et al. DeepSpeed: System Optimizations Enable Training Deep Learning Models with Over 100 Billion Parameters. KDD, 2020.

[22] Schulman, J. et al. Proximal Policy Optimization Algorithms. arXiv:1707.06347, 2017.

[23] Loshchilov, I., and Hutter, F. Decoupled Weight Decay Regularization. ICLR, 2019.

[24] Bommasani, R. et al. On the Opportunities and Risks of Foundation Models. Stanford CRFM, 2021.

خلاصة حاكمية: لا يصبح **MOUSSAIF9B** أقرب إلى **AGI** لأنه يقول أكثر، بل لأنه يعرف ما الفرق الذي يهم، وما السؤال الذي يفتح المسار، وما الذي يحق له فعله، وكيف يهدم قراره، وكيف يعيد بناء نفسه بعد الفعل.