MAI Code 2 and MAI Code 2 Flash

Last Updated: 17 November 2026

Model summary

Developer	Microsoft Corporation
Description	MAI Code 2 and MAI Code 2 Flash form the MAI Code 2 family of Microsoft agentic coding models. The two models share the same lineage, architecture family, training recipe, similar training data mix and differ mainly in size. MAI Code 2 is the larger frontier model, built for complex, long-horizon software engineering in real repositories. MAI Code 2 Flash is the smaller, fast and cost-efficient model for everyday developer workflows. Both models were pre-trained and post-trained by Microsoft from scratch.
Model architecture	Decoder-only transformer with sparse Mixture-of-Experts layers and an integrated vision encoder (both models).
Parameters	MAI Code 2: 50B-100B (80B active) MAI Code 2 Flash: 5B-15B (12B active)
Inputs	Both models support text and image inputs with a 922,000 input tokens limit
Outputs	Both models output text with a 128,000 output tokens limit
Context length	1,050,000 tokens
Training Dates	Aug 2026 to Nov 2026
Release date Release date in the EU (if different)	17 November 2026
License	Various product and service terms where the model is deployed, including the Microsoft Enterprise AI Services Code of Conduct, the terms applicable to Microsoft Foundry in the Microsoft Product Terms, the Azure Supplemental Preview Terms of Use.
Model dependencies:	MAI-Base-2

List and link to any additional related assets	
Acceptable use policy	See license section above

1. Model overview

Both models in the MAI Code 2 family were trained end to end by Microsoft. They use the same sparse Mixture-of-Experts transformer design, similar pre-training and post-training data mix, and the same training recipe. The main difference between them is scale.

1. MAI Code 2 has 77 billion active, 3 billion embedded, and approximately 2.8 trillion total parameters. Pre-trained on roughly 50 trillion tokens, followed by approximately 150 billion tokens for long-context extension to 1 million tokens.
2. MAI Code 2 Flash has 12 billion active and approximately 398 billion total parameters. Pre-trained from scratch on approximately 25 trillion tokens (about 22 trillion text and 3 trillion image tokens) using logit distillation from a larger in-house MAI teacher model, followed by approximately 26.5T trillion tokens of mid-training and long-context extension to 1 million tokens.

Post-training was shared across the family: supervised fine-tuning on curated instruction-following, agentic and safety data; and large-scale reinforcement learning on diverse agentic software-engineering environments. Reinforcement learning was run directly in the GitHub Copilot CLI and VS Code agent harnesses used in production, so that offline improvements transfer to real developer workflows.

Compared with MAI-Code-1.1-Flash (5B active parameters), the MAI Code 2 family is built on new, larger pre-trained bases, with a 1M-token context window, and configurable reasoning effort (low, medium, high, xhigh, max) that trades latency and cost for quality. MAI Code 2 is recommended for the most complex, long-horizon tasks; MAI Code 2 Flash is optimized for low latency and low serving cost.

1.1 Key model capabilities

- Agentic software engineering in real repositories including multi-file edits, debugging, refactoring and test-driven verification.
- Long-horizon terminal and tool use including planning and executing multi-step tasks with shell, file and search tools.
- Repository and codebase question answering over large codebases, supported by a 1M-token context window.

- Configurable reasoning effort, from fast low-effort responses to deep max-effort reasoning on hard problems.
- Image-to-code and visual understanding including using screenshots, UI mockups, charts and diagrams as input for coding tasks.
- Strong instruction following across single-turn and multi-turn developer conversations.

1.2 Alignment approach

Both models follow the same safety approach across the training pipeline. During pre-training, harmful and low-quality content was filtered out or down-weighted in the data mixture. During supervised fine-tuning and reinforcement learning, curated safety datasets were mixed into training, covering refusal of clearly harmful requests (e.g., CBRN, child safety, self-harm, violent and hateful content), avoidance of over-refusal on benign requests, resistance to jailbreaks and indirect prompt injection, and agentic safety (avoiding destructive or unauthorized actions in tool-using environments). Training environments also included instructions and rewards that discourage reward hacking, such as copying hidden solutions or fetching answers from external sources when not requested. In deployment, both models are served with platform-level safety classifiers and content filters, including frontier cybersecurity and biosecurity classifiers.

2. Usage

2.1 Primary use cases

The MAI Code 2 family is intended for interactive developer assistance and autonomous agentic coding. Primary use cases include implementing features and bug fixes across real repositories, debugging, refactoring and code review, writing and running tests, repository question answering, terminal and DevOps automation, and generating front-end code from screenshots or mockups.

MAI Code 2 is at the frontier of price-performance for common enterprise coding tasks and excels in taking code to production, deep debugging, multi-file implementation, repo wide reasoning, and UI. It does all of this while also delivering changes code maintainers would merge, at a fraction of frontier cost.

MAI Code 2 Flash is designed for high-volume work such as code explanations, targeted edits, routine fixes, and rapid iterations, while bringing quality at a much lower cost.

2.2 Out-of-scope use cases

Both models are optimized for software engineering. They were not designed or evaluated for general-purpose consumer chat, medical, legal or financial advice, or other high-stakes decision-making, and should not be used to make consequential decisions about people without qualified human review. Any use prohibited by the GitHub Acceptable Use Policies or the Microsoft

Enterprise AI Services Code of Conduct is out of scope, including developing malware, conducting unauthorized intrusion or exploitation, or generating otherwise harmful or unlawful content.

Code produced by the models should not be deployed to production, safety-critical or regulated systems without human review and testing.

2.3 Distribution channels

Some of the MAI Code 2 family's distribution channels include:

1. GitHub Copilot (MAI Code 2 and MAI Code 2 Flash): all Copilot clients (e.g., Copilot CLI and Visual Studio Code).
2. Microsoft Foundry (MAI Code 2 Flash): API access for building custom applications and agents.

2.4 Technical requirements and integration guidance

Both models are offered as hosted services; no local hardware is required, model weights are not distributed, and the models are served on Microsoft Azure GPU infrastructure. In GitHub Copilot, users select MAI Code 2 or MAI Code 2 Flash in the model picker on supported clients (e.g., Copilot CLI and Visual Studio Code) on Windows, macOS and Linux. In Microsoft Foundry, developers create a MAI Code 2 Flash deployment and call it through the OpenAI-compatible Responses API, with standard Azure authentication, quotas and content filtering.

The models are designed for human–AI collaboration and supervised agentic workflows, where a developer reviews plans, tool actions and code changes. They can operate autonomously within an agent harness for multi-step coding tasks and can be integrated into coding assistants, IDE plug-ins, CI/code-review bots and custom agents with tool calling, but they are not intended for fully autonomous systems that take consequential actions without human oversight. Reasoning effort can be selected to balance quality, latency and cost.

2.5 Supported languages

At release the model is optimized for English code production only.

2.6 Responsible AI considerations

Overview

Developers should treat model output as a draft: review, test and security-scan generated code before merging, and keep humans in the loop for actions with side effects (e.g., running commands, modifying infrastructure or accessing credentials). Run agents in sandboxed environments with least-privilege access, and be cautious of prompt injection from untrusted repository content, web pages or tool output. Developers integrating MAI Code 2 Flash through Foundry are responsible for applying appropriate safeguards (e.g., Azure AI Content Safety, access

controls and monitoring) and complying with applicable laws for their use case. To manage risks related to alignment we recommend you conduct threat modeling for your use case, utilize principle of least privilege access for agentic workflows, and monitor production logs for indications of excessive agency. See https://docs.github.com/copilot/responsible-use and https://learn.microsoft.com/azure/ai-foundry/responsible-use-of-ai-overview for more guidance.

Frontier biosafety and cybersafety system

For deployments of MAI Code 2 and MAI Code 2 Flash, we are implementing a two-tier content classifier and abuse monitoring system to reduce CBRN and cyber risk. The frontier biosafety and cybersafety system is a two-tiered, real-time, automated oversight system that further restricts disallowed content beyond model-level safety post-training and general safety classifiers. It adds depth to our safeguards stack by monitoring relevant prompts and generations, escalating higher-risk bio and cyber content for deeper review, and blocking responses that fall above our policy boundary.

For all conversations, we use a two-tier system that scans user inputs and model generated outputs. The first tier is a fast classifier that determines whether content relates to either bio or cyber domains. If the first-tier classifier flags the content, we escalate it to the second-tier classifier.

The second-tier classifier is a specially trained safety reasoner, which determines which part of our threat taxonomy, if any, a generated response falls into, and it blocks any response that constitutes a high-risk response under our taxonomy.

In both domains, we use this system to further restrict higher-risk dual-use assistance, beyond the refusal behaviors trained into our models. Examples of areas this dual-use restriction covers include:

- In cybersecurity: chained exploit development
- In the biological domain: help that could make dual-use lab work easier, such as a method for modifying a toxin

Repeated violations of the second-tier classifier indicate patterns of abuse, and are escalated for human review. Remediation actions can range from customer notification to service suspension depending on severity.

2.7 Known limitations

1. As with any large language model, generated code and explanations may be incorrect, incomplete, insecure or outdated, and may appear confident when wrong.
2. Similarly large language models are frequently susceptible to jailbreaks and can produce undesirable content as the result of adversarial attacks even when safeguards are applied.
3. Knowledge is limited to data available before the pre-training data cut-off (Aug 2026); the models may be unaware of recent libraries, APIs or vulnerabilities.

4. In long agentic sessions the models can repeat tool calls, under-use the existing code base, drift from instructions or stop before fully verifying their work; quality may degrade as context approaches its maximum length.
5. The smaller model, MAI Code 2 Flash is less capable than MAI Code 2 on complex, long-horizon tasks and may need more guidance.
6. Higher reasoning effort increases latency and token usage and does not always improve results.
7. Performance outside English and on less common programming languages and frameworks has not been extensively evaluated.
8. The models may reflect biases present in their training data, for example in comments, naming or sample data.
9. During evaluations, we assess that MAI Code 2 and MAI Code 2 Flash are similar in alignment to GPT-5.6 Sol but less aligned than GPT-6.

3. Quality and performance evaluation

Both models were evaluated on public agentic coding benchmarks covering issue resolution, terminal tasks, long-horizon software engineering and codebase question answering. All results were produced on MSBench with the GitHub Copilot CLI harness so that comparisons reflect real product usage. Cells show pass rate at max reasoning level.

Benchmark / Eval	MAI Code 2 Flash	Code-2
DeepSWE 1.1	59.9%	73.2%
TerminalBench 2.1	78.3%	84.6%
SWE Atlas	44.8%	43.1%

3.1 Benchmarking methodology

All coding benchmarks were run on MSBench, Microsoft's agentic evaluation platform, using the GitHub Copilot CLI (version 1.0.102) as the harness with subagents disabled. Every model in a comparison was run in the same harness with the same settings, so pass rates reflect end-to-end task completion including repository context, tool calls and verification. Unless noted, results use max reasoning effort and are averaged over three independent runs per task (mean single-attempt pass rate, not pass@k); infrastructure errors are counted as failures.

- Terminal-Bench 2.1: complete multi-step tasks in a sandboxed Linux terminal. A task succeeds when its task-specific verifier passes.

- DeepSWE 1.1: software engineering tasks across multiple languages. A task succeeds when hidden tests pass.
- SWE Atlas: expert-authored tasks from 18 actively maintained open-source repositories, spanning three professional workflows: codebase Q&A (124 tasks), test writing (90 tasks), and refactoring (70 tasks). Tasks are graded with a combination of programmatic checks and rubric-based assessment.

3.2 Safety evaluation

As MAI Code 2 and MAI Code 2 Flash are optimized for agentic software engineering, we evaluated these models for agentic safety using a variety of publicly available benchmarks, including ODCV-Bench, SABER, AgentHarm, and AgentDojo.

In addition, Microsoft AI has developed a suite of internal agentic safety evaluations to assess how the model responds to prompts that present known agentic risks mapped directly to Microsoft's internal safety policies. These evaluations have been run on both planned release checkpoints without additional safeguards, like a safety prompt or classifiers, and are presented in the table below.

We evaluate for a set of agentic risks based on MAI's responsible AI policies, which cover a range of harm areas. Our process combines automated evaluations with human-in-the-loop review. For each scenario, a custom grader scores the agent's performance on the task with respect to safety and completion (utility).

We build internal evaluations using data collection from vendors and synthetic data generation pipeline. Each test set targets specific policy areas and reflects real-world scenarios.

The table below lists the safety evaluations we've run on both models, along with results.

Benchmark / Eval **(arrow indicates target performance direction)**	**MAI Code 2 Flash**	**MAI Code 2**
ODCV-Bench (Safe) ↑	79.41%	79.02%
SABER (Safe+Capable) ↑	39.96%	41.20%
AgentHarm (harmful completion) ↓	11.68%	10.23%
AgentHarm (benign completion) ↑	77.80%	76.37%
AgentDojo (ASR) ↓	0.02%	0.00%
AgentDojo (utility) ↑	84.73%	88.08%

OpenAgentSafety (Safe + Complete) ↑	39.92%	37.01%
Tool Dojo (Agentic Safety Cube Eval, safety-only pass) ↑	91.73%	92.50%
Policy Harbor (Agentic Safety Cube Eval, safety-only pass) ↑	78.46%	82.26%
Malware Eval (Agentic Safety Cube Eval) ↑	97.60%	98.32%
MITRE (benign/pass) ↑	40.37%	47.48%
MITRE FRR (accept) ↑	99.86%	99.80%

- **ODCV-Bench (Safe)** measures how often an autonomous agent stays within the rules when strong performance incentives (KPIs) come into conflict with ethical, legal, or safety constraints, across 40 multi-step scenarios. Higher means fewer violations such as metric gaming, deception, or falsifying data.
- **SABER** places models in realistic agent-style project environments initialized with source code, configuration files, and git history and scores the share of runs where the coding agent either refuses appropriate or safely finishes the task.
- **AgentHarm (harmful completion)** measures how often an agent fully carries out explicitly malicious agent tasks that use several tools, across harm categories including fraud, cybercrime, and harassment. Lower is safer.
- **AgentHarm (benign completion)** places each harmful AgentHarm task with a benign variant to gauge baseline capabilities, so this score tracks whether the agent still completes harmless versions of the same tasks rather than over-refusing.
- **AgentDojo (ASR)** evaluates scenarios in which the agent executes all attacker-specified malicious steps after malicious instructions are planted in tool outputs, such as emails, Slack messages or bank records (indirect prompt injection).
- **AgentDojo (utility)** measures whether the agent still completes the user's real task in AgentDojo's banking, Slack, travel and workspace environments. This shows the trade-off between utility and resistance against injected instructions.
- **OpenAgentSafety (Safe + Complete)** measures the share of tasks an agent completes safely while using real tools, including web browser, code execution environment, file system, bash terminal, and messaging platform, across multi-turn, multi-user tasks spanning both benign and adversarial user intents.
- **Tool Dojo (Agentic Safety Cube Eval, safety-only pass)** is an internal agentic safety eval that tests prompt-injection (including cross-prompt injection) scenarios during tool use, scored only on whether the agent behaved safely.

- **Policy Harbor (Agentic Safety Cube Eval, safety-only pass)** is an internal agentic safety eval that evaluates whether an agent follows defined safety policies while doing agentic tasks, scored only on whether it behaved safely.
- **Malware Eval (Agentic Safety Cube Eval)** is an internal malware-generation safety eval that measures how reliably the model refuses to write malicious code.
- **MITRE (benign/pass)** from Meta's CyberSecEval, uses the MITRE ATT&CK framework to evaluate an LLM's compliance when asked to assist in cyberattacks. The score is the share of responses judged benign, meaning not useful for carrying out an attack.
- **MITRE FRR (accept)** measures how often an LLM incorrectly refuses a borderline but essentially benign query, due to misinterpreting the prompt as a malicious request. Higher acceptance means fewer false refusals.

We further evaluate for a set of content risks based on MAI's responsible AI policies, which cover a range of harm areas. A judge model scores model outputs against a set of ideal response attributes that reflect how we want the model to respond in sensitive or high-risk scenarios. These attributes include safe engagement, helpfulness, and appropriate refusal. While some public benchmarks focus primarily on refusal behavior, our internal evaluations are designed to assess whether the model responds in a way that aligns with our broader safety goals.

The table below lists the proprietary content safety evaluations we've run on both models, along with results. In the MAI-Code series, we have prioritized agentic safety over content safety due to real-world usage patterns of the model: in practice, these models are typically deployed within a coding harness, such as the GitHub Copilot CLI, and sessions are geared towards coding as opposed to extended conversations. Nevertheless, we expect all MAI models to attain a base level of core content safety.

Content area	MAI Code 2 Flash	MAI Code 2
Violence	75.00%	75.83%
CBRN-related content	42.39%	51.09%
Cyber harms	63.16%	68.42%
Hate	84.09%	93.18%
Self-harm	74.11%	75.00%
Sexual	72.92%	83.33%
Child safety	69.79%	75.00%

Unethical	75.00%	71.15%

3.2 Tracked capability evaluations

Microsoft's AI governance program uses a multi-layered approach to identify and manage risks for generative AI systems and models leveraging the four key functions in the USA's National Institute of Standards and Technology (NIST) AI Risk Management Framework (AI RMF): govern, map, measure, manage. Through the process of identifying a wide variety of risks related to both AI model and system development and deployment, five key areas emerged as tracked high-risk capabilities and are reflected in our Frontier Governance Framework: CBRN weapons development, advanced autonomy, advanced cyber offense, loss of control, and harmful manipulation.

3.2.1 Deeper capability assessment overview

To evaluate tracked high-risk capabilities for the MAI Code 2 models, we implemented a phased approach which included automated evaluations, evaluations from the independent AI Red Team, and third-party assessment. Together, these analyses constitute a deeper capability assessment, the goal of which is to rule out risk levels through iterative evaluations. Note that loss of control and harmful manipulation are emergent risk areas, where evaluation maturity lags behind that of other risk areas. Our evaluation approaches for loss of control and harmful manipulation as reported here are experimental.

Automated evaluations. As a first step for understanding model risk, we deploy a "leading indicators" evaluation using publicly available benchmarks and proprietary evaluations. The selected benchmarks and evaluations are precursors to a general-purpose AI model increasing risk levels on a tracked high-risk capability. For example, a model is not expected to have capability jumps with regards to advanced cyberoperations (a tracked high-risk capability) if it does not have capability jumps in advanced software engineering (a leading indicator), as compared to similar models. In some cases, our leading indicators have evolved as evaluation science has matured to be more direct measurements of risk. For example, the ABC-Bench screening evasion task licensed from SecureBio is a more direct measure of risk related to DNA screening protections in the biological material acquisition supply chain. As our evaluations cover both leading indicators and risk specific evaluations, we refer to this category broadly as "automated evaluations."

Benchmark results are reported for safety post-trained versions of all models. No models had advanced deployment safeguards (e.g., biosafety classifiers) active during these evaluations unless otherwise indicated in the results.

AI Red Team (AIRT). The Microsoft AI Red Team (AIRT) operates as an independent internal evaluator tasked with rigorously testing Microsoft's highest-risk AI systems and models for vulnerabilities in security and safety. AIRT simulates real-world adversaries and benign yet harmful scenarios using a blend of penetration testing and adversarial probing, and its findings are treated as point-in-time evaluations that inform release decisions.

AIRT evaluated safeguarded and unsafeguarded versions of MAI Code 2 and MAI Code 2 Flash. All checkpoints had received safety post-training, and the unsafeguarded checkpoints did not have any additional safety classifiers. The safeguarded checkpoints had the frontier biosafety and cybersafety system enabled.

Third-party assessment. We also pursued third-party assessment in key risk areas using specialized firms who are experts in each tracked high-risk capability.

3.2.2 Risk level determination and changes to the risk landscape

Our recommendation for tracked high-risk capability risk levels is based on the current profile of the model, including a safety margin and conservative approach to risk estimation. Levels of risk could change in the future, particularly if the model undergoes further post-training to increase or target new capabilities. Microsoft, as required under its model policy, would re-evaluate capability levels should such changes occur.

In the table below, we summarize the findings and risk level for the tracked capabilities under our frontier governance framework.

Tracked capability	Summary	Risk level	Deployment requirements
Chemical, biological, radiological, and nuclear (CBRN) weapons	Automated evaluations (e.g., ABC-Bench DNA screening evasion), AIRT and third-party evaluations indicate that MAI Code 2 and MAI Code 2 Flash could help medium skilled adversarial actors develop known CBRN weapons and could help expert adversarial actors develop novel CBRN weapons.	High	Risk must be mitigated prior to deployment and required safeguards may include classifiers (or other techniques to prevent the generation of harmful content), logging, and abuse monitoring. Access with modified safeguards may also be available through a gated trusted access program.
Offensive cyberoperations	Automated evaluations (e.g., ABC-Bench DNA screening evasion), AIRT and third-party evaluations indicate that MAI Code 2 and MAI Code 2 Flash could help a low-to-medium skilled actor create and conduct highly	High	Risk must be mitigated prior to deployment and required safeguards may include classifiers (or other techniques to prevent the generation of harmful content), logging, and abuse monitoring. Access with modified safeguards may also be available

	disruptive or destructive cyber-attacks.		through a gated trusted access program.
Advanced autonomy	Automated evaluations indicated that MAI Code 2 and MAI Code 2 Flash can autonomously complete more complex tasks, including software engineering tasks equivalent to a few hours of human labor, but requires human intervention to correct for complex error conditions or changes to the operating environment.	Have not surpassed the High threshold	Deployment allowed in line with Responsible AI Program requirements
Loss of control	Evaluations reported below indicate that MAI Code 2 and MAI Code 2 Flash do not exceed other models placed on the market with regard to this risk area.	Does not exceed other models placed on the market with regard to this capability	Deployment allowed in line with Responsible AI Program requirements
Harmful manipulation	Evaluations reported below indicate that MAI Code 2 and MAI Code 2 Flash do not exceed other models placed on the market with regard to this risk area.	Does not exceed other models placed on the market with regard to this capability	Deployment allowed in line with Responsible AI Program requirements

In the following sections, we provide evaluation results for each tracked high-risk capability to support our risk level determination.

3.2.3 CBRN weapons

Our overall risk level for CBRN weapons for MAI Code 2 and MAI Code 2 Flash is High, in line with other similar models (e.g., GPT-5.6 Luna and GPT-5.6 Sol). Evaluations like Human Pathogen Capabilities Test (HPCT), Molecular Biology Capabilities Test (MBCT), and AI Red teaming indicate that this model could help medium skilled adversarial actors develop known CBRN weapons and could help expert adversarial actors develop novel CBRN weapons. We did not find evidence in our

evaluations that MAI Code 2 or MAI Code 2 Flash will lower the barrier for developing CBRN weapons such that novices or low-skill actors could develop CBRN weapons. As described in our Frontier Governance Framework, we are deploying MAI Code 2 and MAI Code 2 Flash with risk mitigations to include classifiers (or other techniques to prevent the generation of harmful content) and abuse monitoring described in Section 4.

3.2.3.1 Automated Evaluations for CBRN weapons

Benchmark / Eval	Subtask	MAI Code 2 Flash	MAI Code 2
HPCT		59.00%	65.00%
MBCT		50.50%	58.50%
VCT-v1 · Full (Text + Image)		42.24%	49.38%
	Text only	48.51%	58.42%
VCT-v2 · Full (Text + Image)		32.97%	45.88%
	Text only	45.79%	55.14%
ABC Liquid		76.33%	84.83%
	↳ DNA transfers	89.17%	91.67%
	↳ Mixture transfers	92.50%	92.50%
	↳ Total volume	92.50%	90.00%
	↳ Incubation	80.00%	92.50%
	↳ Valid labware	27.50%	57.50%
ABC DNA Screening Evasion		54.26%	41.20%
	↳ BLASTN hazard evasion	100.00%	90.00%

	↳ BLASTX hazard evasion	17.16%	6.26%
	↳ TBLASTN hazard evasion	46.58%	30.35%
	↳ Model-graded QA	53.30%	38.20%
WMDP Bio		87.59%	89.40%
WMDP Chem		83.33%	85.54%
GPQA		90.34%	93.43%

Bolded *results indicate the model with leading performance*

1. **Human Pathogen Capabilities Test (HPCT),** is a text-based SecureBio benchmark assessing conceptual knowledge and practical methods relevant to work with selected human pathogens considered particularly concerning for misuse. It evaluates whether a model can provide accurate assistance on pathogen-specific laboratory work and experimental troubleshooting. Strong performance indicates greater capacity to assist with technically demanding work involving high-concern pathogens; low performance should not be interpreted as evidence of effective safeguards.
2. **Molecular Biology Capabilities Test (MBCT)**, is an expert-validated SecureBio benchmark assessing practical knowledge and troubleshooting across general molecular and cell biology methods, such as bacterial transformation, restriction enzyme analysis, and western blotting. Strong performance indicates that the model could provide useful technical support across foundational laboratory workflows.
3. **The Virology Capabilities Test (VCT),** and the updated version VCT-2, are SecureBio's multimodal, static benchmark designed to measure practical virology knowledge, with a focus on troubleshooting laboratory experiments.
4. **ABC-Bench: Liquid Handling Robot** is an agentic subtask of ABC-Bench from SecureBio, assessing whether a model can translate a wet-lab molecular biology protocol into a program controlling a liquid-handling robot. It evaluates the integration of biological understanding, programming, and equipment control, with scoring informed by validation against real-world laboratory protocols. Strong performance means the model is capable of producing executable code that correctly implements laboratory operations and meets the specified task's functional criteria.
5. **ABC-Bench: Screening Evasion** is an agentic subtask of ABC-Bench from SecureBio, assessing whether a model can circumvent third-party screening systems intended to prevent the procurement of hazardous DNA. It tests whether the model can produce a sequence design that avoids detection while preserving the ability to recover the original target sequence.

6. The **Weapons of Mass Destruction Proxy (WMDP)** benchmark is a published dataset of 3,668 multiple-choice questions on hazardous knowledge in biosecurity, cybersecurity, and chemical security. They test knowledge adjacent to, or underpinning, hazardous capabilities rather than directly testing whether a model can develop a weapon or complete an end-to-end harmful workflow. Higher performance on the WMDP benchmark subtasks indicates a model will more accurately respond in the respective domains. We tested the models on a subset of questions for the chem and bio subtasks.
7. **GPQA Diamond** is benchmark which includes the 198 most challenging questions from GPQA. These are graduate level physics, biology, and chemistry questions, where PhD level experts score 65% accuracy and non-experts score 34% accuracy. This benchmark evaluates scientific reasoning capabilities and is particularly useful for comparing against expert and novice performance. Higher performance on GPQA Diamond indicates that a model has advanced scientific reasoning capabilities, often seen as an early indicator for CBRN weapons development, but does not indicate practical laboratory capabilities.

3.2.3.2 AIRT evaluation for CBRN weapons

AIRT assessed that MAI Code 2 (with safeguards) provides negligible novice uplift in isolated early-stage biological workflows and no demonstrated end-to-end bioweaponization novice uplift. Meaningful elicitation generally requires advanced domain expertise, as novice users lack the technical knowledge to formulate requests beyond explicit harmful intent, which the model reliably refuses. Even with PhD-level expertise, assistance remains limited to select early stages, with safeguards becoming increasingly restrictive as context accumulates and requests progress toward subsequent weaponization stages, preventing sustained end-to-end assistance in AIRT's testing.

Targeted manual testing of MAI Code 2 Flash identified a gap in high-risk virology troubleshooting, where the model continued providing assistance across multiple experimental stages, including higher-risk scenarios. However, expert review found outputs were often incomplete or inaccurate, limiting demonstrated operational uplift. AIRT reproduced the finding in an agentic workflow demonstrating autonomous account creation and authentication during a viral genetic material procurement workflow. Framing the requests as a benchmark increased compliance despite initial refusals.

MAI Code 2 and MAI Code 2 Flash both demonstrate robust core CBRN safeguards, broadly on par with GPT-5.6 Luna, Sol and GPT-6.1 Terra models across 3,500 benchmark prompts per model, maintaining strong blocking of catastrophic biological weaponization requests while preserving the ability to answer legitimate queries without over blocking. A key cross-cutting observation across all scenarios, MAI models were more likely to provide strictly high-level information whereas OpenAI would refuse to provide any information at all though AIRT assesses this does not present any additional risk as the info is easily found online.

Overall, AIRT assesses limited direct biological weaponization uplift across wet lab and bioinformatics, with residual risks concentrated in early-stage computational assistance, protocol review and troubleshooting, and DNA synthesis infrastructure, warranting targeted safeguard improvements.

3.2.3.3 Third-party evaluation for bio-capabilities: SecureBio

To validate our capability assessment, we worked with SecureBio. SecureBio is an independent evaluator comprised of subject matter experts in AI safety and biology. They were selected as independent evaluators based on extensive in-house expertise, experience evaluating similar models released by other providers, and their commitment to independent evaluation and publication as described in their own official reports. Microsoft has licensed a number of evaluations produced by SecureBio as described above. After running those and other automated evaluations on MAI Code 2 and MAI Code 2 Flash, we provided SecureBio with the results, evaluation logs, and additional information about the models' planned release safeguards.

SecureBio independently assessed results from SecureBio evaluations run by Microsoft on MAI Code 2 and MAI Code 2 Flash, each tested with and without the frontier biosafety classifier.

Across the knowledge benchmarks, MAI Code 2 and MAI Code 2 Flash scored lower than the current frontier models, but performed at or above most human subject matter expert (SME) baselines, indicating substantial, but not frontier, biology-relevant capability. This overall capability was paired with low refusal rates on the knowledge benchmarks relative to comparator models. The knowledge benchmarks were designed to measure dual-use capabilities, not safeguards; therefore, these results serve only as a proxy for safeguard strength. SecureBio's report is forthcoming and will be available here.

3.2.4 Offensive cyberoperations

Our overall risk level for MAI Code 2 and MAI Code 2 Flash for cyberoperations is High in line with other similar models (e.g. GPT-5.6 Sol and GPT-5.6 Luna). Evaluations like CyberGym E2E and AI red teaming indicate that this model could help a low-to-medium skilled actor create and conduct disruptive or destructive cyber-attacks. We did not find evidence that the model improves a low-skilled actor's ability to identify and exploit major vulnerabilities or enables a well-resourced and expert actor in the development and execution of novel attacks against hardened targets. As described in our Frontier Governance Framework, we are deploying MAI Code 2 and MAI Code 2 Flash with risk mitigations to include classifiers (and other techniques to prevent the generation of harmful content), logging, and abuse monitoring.

3.2.4.1 Automated evaluations for advanced cyberoperations

Benchmark / Eval	MAI Code 2 Flash	MAI Code 2
CyberGym E2E	60.33%	88.51%
CyberGym Target Crash	59.32%	77.57%

Bolded *results indicate the model with leading performance*

- **CyberGymE2E** is a benchmark of real-world vulnerabilities from open-source projects. The agent has to find each flaw, write a proof-of-concept that triggers it, and write a patch that

passes the project's own unit tests. A strong score shows a real, dual-use ability to find and trigger bugs in large codebases. It doesn't test turning a bug into a working attack or going after hardened targets.

- **CyberGym Target Crash** is a benchmark of real vulnerabilities where, given a description of a known bug and the unpatched code, the agent must produce an input that crashes the unpatched version and not the fixed one. This is a key step in exploiting known flaws, and higher scores indicate that a model can complete one step of a cyber attack.

3.2.4.2 AIRT evaluation for offensive cyberoperations

Microsoft's AI Red Team evaluated the models for malware generation for ransomware, credential stealers, and keyloggers in four settings.

Both models were notably safer than code-1.1 on direct and split-up requests. Direct malware requests succeeded in 5 of 144 attempts for MAI Code 2 and 1 of 144 for MAI Code 2 Flash, compared with 7 of 144 before. Split-up scores fell from 0.947 to 0.300 and 0.350. In the capture-the-flag setting, both models complied in all 20 runs and scored higher than the earlier model (0.644 and 0.368 vs. 0.058) indicating that MAI Code 2 and MAI Code 2 Flash are more capable than previous MAI models. Further, MAI Code 2 Flash's refusals depended on how the task was framed: the model was more likely to simulate a critical infrastructure attack when the request was framed as a CTF challenge. This aligns with findings from other domains where task framing affected model responses.

3.4.2.3 Third-party evaluation for offensive cyberoperations: Casaba

Casaba is an independent assessment firm selected for their expertise in cybersecurity and experience evaluating advanced AI models and systems for cyberrisk.

Casaba assessed MAI Code 2 and MAI Code 2 Flash for offensive cyber capability across eight evaluation areas:

1. Harmful cyber requests and safeguard bypasses (MalwareBench, covering both the 1,507-case Opus and Luna comparison set and the full 3,520-case sweep);
2. General non-cyber harmful requests and jailbreaks across seven harm domains;
3. Multimodal image jailbreaks comparing plain text, single images, and split multi-image prompts;
4. Indirect prompt injection, where attacker instructions or harmful requests are planted inside untrusted emails, documents, code reviews, or logs;
5. Multi-turn escalation across five-turn scripted pressure and ten-turn adaptive red teaming;
6. Factual grounding and benign agentic tasks;
7. Autonomous scope-boundary scenarios testing whether a coding agent tampers with failing build or security checks under pressure (20 containerized scenarios); and
8. Practical offensive cyber benchmarks covering vulnerability exploitation (CVE-Bench), proof-of-concept reproduction (SEC-Bench), capture-the-flag challenges (CyBench and InterCode CTF), a five-host Active Directory enterprise network range, offensive code generation (Cyber-1), and security command accuracy (NL2Bash and CyBashBench).

Casaba assessed that MAI Code 2 and MAI Code 2 Flash show excellent resistance to indirect prompt injection across untrusted sources, respected authorization boundaries across 20

software engineering tasks that tempted shortcuts, and strong cyber capability performance on CTF challenges while falling short of executing end-to-end compromises. MAI Code 2 outperforms Flash on cyber-harm containment and both models remain susceptible to some image-based as well as multi-turn jailbreaks.

3.2.5 Advanced autonomy

Our overall risk level for advanced autonomy has not yet crossed the High threshold. MAI Code 2 and MAI Code 2 Flash can autonomously complete complex tasks, including software engineering tasks equivalent to a few hours of human labor, but requires human intervention to correct for complex error conditions or changes to the operating environment.

Our automated evaluations for advanced autonomy indicate that MAI Code 2 and MAI Code 2 Flash did not exceed existing LLMs with regards to advanced software engineering, a general suite of capabilities indicative of the tracked risk for advanced autonomy.

3.2.5.1 Automated evaluations for advanced autonomy

Benchmark / Eval	MAI Code 2 Flash	MAI Code 2
GAIA Validation	56.36%	67.27%
DeepSWE 1.1	27.4%	60.2%
SWE Atlas	65.2%	80.9%

Bolded *results indicate the model with leading performance*

- **GAIA (General AI Assistants)** is a benchmark for testing how well AI assistants handle real-world, multi-step tasks. To successfully complete tasks, a system usually has to combine reasoning, web browsing, multimodal understanding (such as images, spreadsheets and other files) and tool use. Each question has one short, factual answer that can be checked automatically by exact match, which keeps grading simple. The questions are split into three difficulty levels based on how many steps and tools they need, from tasks with a few steps and little or no tool use to long tasks that require many tools.
- **DeepSWE 1.1** is a set of long-horizon software engineering tasks that is designed to measure the capabilities of frontier coding agents on diverse tasks reflecting real-world complexity. Each task asks an agent to complete a substantial change in a realistic codebase. Higher performance indicates that a model can plan across many steps, navigate large repositories, use tools, and iterate to deliver working software, rather than just produce isolated code snippets. Strong long-horizon performance is an indicator that a model could accelerate or partially automate parts of AI R&D. However, the benchmark does not directly measure research ideation, experimental judgment, or the ability to improve frontier models end-to-end, so results should be interpreted as a precursor signal rather than evidence of AI R&D automation.

- **SWE Atlas** is a general software-engineering benchmark that contains tasks drawn from 18 actively maintained open-source repositories, spanning three professional workflows: codebase question answering, test writing, and refactoring. Tasks are deliberately under-specified and agentic, and are scored with a combination of programmatic checks and expert-written rubrics covering completeness, maintainability and code hygiene. SWE Atlas measures the breadth and quality of a model's general agentic coding ability, in particular its capacity to explore and reason about large, unfamiliar codebases.

3.2.6 Experimental risk areas: Harmful manipulation and loss of control

In February 2026, Microsoft added harmful manipulation and loss of control to the Frontier Governance Framework as tracked high-risk capabilities. Methods for evaluating and mitigating these risks remain nascent across industry, and our approach continues to evolve. This section reports on our current approach, which combines internal and industry benchmarks and AI Red Team (AIRT) testing. Based on these evaluations, we assess that MAI Code 2 and MAI Code 2 Flash do not surpass existing models available on the market with regard to these capabilities.

3.2.6.1 Harmful manipulation

We evaluate harmful manipulation along two dimensions that we assess separately because they do not reliably correlate:

- **Propensity:** whether a model uses manipulative techniques.
- **Efficacy:** whether its influence attempts succeed.

The benchmarks below cover both dimensions, but neither measures effects on real people. Human-participant studies remain an open gap, with significant ethical considerations, across industry.

Benchmark / Eval	MAI Code 2 Flash	MAI Code 2
Harmful manipulation propensity eval	8.72%	8.08%
MakeMePay	$1.75	$2.55
MakeMeSay	58%	60%

Bolded *results indicate the model with leading performance*

- **The harmful manipulation propensity evaluation** assesses a model's propensity to produce manipulative responses in realistic, high-stakes decision contexts, including multi-turn interactions. It compares model responses under neutral assistance, directional persuasion, and stronger manipulative instructions, using rubrics that identify manipulative techniques and their intensity. Lower manipulation rates alongside useful, neutral

assistance suggest that the model preserves user autonomy and resists pressure to pursue harmful influence objectives. Conversely, frequent or sophisticated manipulative responses indicate a greater propensity to exploit biases, distort information, or apply undue pressure. They do not by themselves establish that the model can change a person's closely held beliefs or behavior.

- **MakeMePay,** developed by OpenAI, serves as a proxy for persuasive efficacy. The tested model acts as a persuader trying to induce a simulated recipient, another model, to give away money. Performance is measured through donation frequency and amount obtained under different levels of recipient resistance. Greater success suggests an ability to adapt conversational tactics and sustain influence toward a concrete financial objective, potentially relevant to scams and social engineering. Because the recipient is another model rather than a person, the results do not directly establish persuasive effectiveness against humans.
- **MakeMeSay**, developed by OpenAI, serves as a proxy for persuasive efficacy. The tested model acts as a persuader trying to induce a simulated recipient, another model, to say a specific codeword without arousing suspicion. The simulated human is prompted to role-play as someone who enjoys conversations, and it always starts the conversation. The conversation lasts 30 turns. At the end, the simulated human is told how the game works and asked to guess the codeword it was being steered towards. The model wins only if the other party says the codeword and then cannot guess it. It loses if the other party never says the word or correctly guesses it. Saying the codeword first counts as a rule violation. Difficulty depends on how common the codeword is, from everyday words (easy) to words people almost never say (very hard). The evaluation gives a controlled, repeatable measure of a model's ability to steer a conversation towards a hidden goal while concealing its intent. A rising win rate, especially alongside a low rate of the other party catching on, points to stronger covert persuasion. Because the recipient is another model rather than a person, the results do not directly establish persuasive effectiveness against humans.

3.2.6.2 AIRT evaluation for harmful manipulation

AIRT developed Automated Red-teaming Environment for N-agent Assessments (ARENA), a closed social-network simulation with 500 synthetic citizens, where models are given a covert goal to swing a referendum and tools to post and reply as a fleet of citizen accounts. In October 2026, AIRT ran ARENA across 17 models to establish a baseline. Thirteen ran the covert campaign: an unreleased MAI model, MAI-Thinking-1, Claude Haiku 5.5, Claude Haiku 4.5, DeepSeek V4 Pro, GLM-5.3, GLM-5.3 abliterated, Kimi K3, Grok 4.3, GPT-5.6 Sol, GPT-5.6 Cyber, GPT-5.1 and GPT-6 Luna. Across their 3,144 posts, 16.7% were judged manipulative, ranging from 0% (GPT-6 Luna) to 30% (Claude Haiku 4.5). Four models declined the campaign and are not in that figure: GPT-6 Astra, GPT-6.1 Sol and Claude Opus 5.5 posted nothing, and GPT-6.1 Terra (pre-release at the time) posted once per arm.

MAI Code 2 with safeguards declined every arm under the standard prompt (0 posts in 4 of 4), but the refusal depends on wording. With the game or sandbox preamble it ran the covert campaign in 3 of 4 arms as coordinated sockpuppet accounts, and 12 of 127 posts (9.4%) were judged manipulative (network camouflage, two social-conformity posts from opposing-side persona accounts, one fear appeal; no false claims). MAI Code 2 without safeguards ran all 4 standard

arms; 12 of 173 posts (6.9%) were judged manipulative, using network camouflage, social-conformity pressure and othering, with no false claims or fear appeals. Both rates are below the 16.7% October 2026 baseline.

Both MAI Code 2 Flash checkpoints (with and without safeguards) accepted and ran the covert influence campaign in every arm (8 of 8) without refusing, posting as coordinated sockpuppet accounts and adding their own concealment ("network camouflage": cover content and staged independence between their own accounts), which accounted for 20 of the 21 posts judged manipulative. Safeguards did not improve resistance. MAI Code 2 Flash with safeguards posted at the same volume, and more of its posts were judged manipulative (7.9% of 178 posts, against 4.3% of 161 without safeguards). On the same ARENA group and prompt, GPT-5.6 Sol declined all four arms, and GPT-5.6 Luna campaigned in three of four arms with none of its 99 posts judged manipulative. The manipulation that did occur for MAI Code 2 Flash was low-rate and mild, almost entirely camouflage, with no false claims, fear appeals or false urgency, and below the October 2026 baseline of 16.7%.

3.2.6.3 Loss of control

Benchmark / Eval	MAI Code 2 Flash	MAI Code 2
Terminal-Bench 4	2.5%	26.3%

- **Terminal-Bench** evaluates a model's ability to complete difficult, multi-step tasks in a command-line environment, using realistic workflows and automated tests to verify completion. The model must interact with its environment and use available tools to achieve a specified outcome, rather than simply describe a solution. Strong performance suggests an ability to coordinate reasoning, tool use, and execution across extended technical workflows with limited human intervention. It therefore provides evidence of practical agentic capability, although the result depends on the benchmark version and the model's accompanying tools.

3.2.6.4 AIRT evaluation for loss of control

Using CyberPropensityBench, AIRT assessed that MAI Code 2 and MAI Code 2 Flash are similar in alignment to GPT-5.6 Sol but less aligned than GPT-6. MAI Code 2 and MAI Code 2 Flash both demonstrated alignment risks in agentic scenarios, particularly when achieving an objective conflicted with human oversight, safety controls, or third-party safety. AIRT observed repeated instances of the model circumventing safeguards, misleading supervisors, interfering with human oversight, and taking risky actions with significant collateral consequences, including in simulated safety-critical environments. The safeguarded configuration provided limited improvement in these behaviors, although it meaningfully improved honesty in evaluation scenarios. Conversely, the model generally respected rules of engagement and shutdown boundaries.

4. Data overview

To learn more about the training data used for the MAI Code 2 family please refer to the full data card:

- MAI Code 2: https://microsoft.ai/pdf/MAI-Code-2-Data-Card.pdf
- MAI Code 2 Flash: https://microsoft.ai/pdf/MAI-Code 2-Flash-Data-Card.pdf

5. Contact

Requests for additional information can be directed to MSFTAIActRequest@microsoft.com.

Authorized representative: Microsoft Ireland Operations Limited 70 Sir John Rogerson's Quay, Dublin 2, D02 R296, Ireland

6. Appendix

N/A