

Xiaomi releases MiMo-V2.5 open omnimodal models under MIT licence

Source copy kept by wheresthe.ai for the story "Xiaomi releases MiMo-V2.5 open omnimodal models under MIT licence" (<https://wheresthe.ai/d/cn-xiaomi-mimo-v2-5-omnimodal-2026-04/>).

Source: <https://huggingface.co/XiaomiMiMo/MiMo-V2.5>

Text saved 18 Oct 2026 (the page as published).

SHA-256 of the saved text: dd303d3181330dbfd79e0b628094e22f1ce8d889e1754878bd425aee407b0b44

```
' ~ (json_dict[json_key] | tojson | safe) ~ '<' ~ json_key ~ '>' }}\n {%- else %}\n {{- '\n<' ~ json_key ~ '>' ~  
(json_dict[json_key] | string) ~ '<' ~ json_key ~ '>' }}\n {%- endif %}\n {%- endfor %}\n {%- endif %}\n {%- endmacro  
-%}\n {%- macro render_content(message_content) -%}\n {%- if message_content is string -%}\n {{-  
message_content -}}\n {%- else -%}\n {%- for content in message_content -%}\n {%- if content['type'] == 'image' or  
'image' in content or 'image_url' in content -%}\n {{- '<|vision_start|><|image_pad|><|vision_end|>' -}}\n {%- elif  
content['type'] == 'audio' or 'audio' in content or 'audio_url' in content -%}\n {{- '<|mimo_audio_start|><|audio_pad|  
><|mimo_audio_end|>' -}}\n {%- elif content['type'] == 'video' or 'video' in content or 'video_url' in content -%}\n {{-  
'<|vision_start|><|video_pad|><|vision_end|>' -}}\n {%- elif 'text' in content -%}\n {{- content['text'] -}}\n {%- endif -  
%}\n {%- endfor -%}\n {%- endif -%}\n {%- endmacro -%}\n {%- if messages[0][\"role\"] == \"system\" %}\n {%- set  
system_message = messages[0][\"content\"] %}\n {%- set loop_messages = messages[1:] %}\n {%- else %}\n {%-  
set loop_messages = messages %}\n {%- endif %}\n {%- set ns = namespace(last_user_index=-1) %}\n {%- for m in  
loop_messages %}\n {%- if m.role == 'user' %}\n {%- set ns.last_user_index = loop.index0 -%}\n {%- endif  
%}\n {%- endfor %}\n {%- if not tools is defined %}\n {%- set tools = [] %}\n {%- endif %}\n {%- if system_message is  
defined %}\n {{- '\n<|im_start|>system\n\n' + render_content(system_message) }}\n {%- else %}\n {{- '\n<|im_start|  
>system\n\nYou are MiMo, a helpful AI assistant engineered by Xiaomi.\n' }}\n {%- endif %}\n {%- if tools is iterable  
and tools | length > 0 %}\n {{- '\n\n\n# Tools\n\n\nYou may call one or more functions to assist with the user query.\n\n\nYou have access to the following functions:\n\n\n' }}\n {{- '\n<tools>' }}\n {%- for tool in tools %}\n {%- if  
tool.function is defined %}\n {%- set tool = tool.function %}\n {%- endif %}\n {{- '\n\n<function>\n\n<name>' ~  
tool.name ~ '\n</name>' }}\n {%- if tool.description is defined %}\n {{- '\n\n<description>' ~ (tool.description | trim) ~  
'\n</description>' }}\n {%- endif %}\n {{- '\n\n<parameters>' }}\n {%- if tool.parameters is defined and tool.parameters  
is mapping and tool.parameters.properties is defined and tool.parameters.properties is mapping %}\n {%- for  
param_name, param_fields in tool.parameters.properties.items %}\n {{- '\n\n<parameter>' }}\n {{- '\n\n<name>' ~  
param_name ~ '\n</name>' }}\n {%- if param_fields.type is defined %}\n {{- '\n\n<type>' ~ (param_fields.type | string)  
~ '\n</type>' }}\n {%- endif %}\n {%- if param_fields.description is defined %}\n {{- '\n\n<description>' ~  
(param_fields.description | trim) ~ '\n</description>' }}\n {%- endif %}\n {%- set handled_keys = ['name', 'type',  
'description'] %}\n {{- render_extra_keys(param_fields, handled_keys) }}\n {{- '\n\n</parameters>' }}\n {%- endfor  
%}\n {%- endif %}\n {%- set handled_keys = ['type', 'properties'] %}\n {{- render_extra_keys(tool.parameters,  
handled_keys) }}\n {{- '\n\n</parameters>' }}\n {%- set handled_keys = ['type', 'name', 'description', 'parameters']  
%}\n {{- render_extra_keys(tool, handled_keys) }}\n {{- '\n\n</function>' }}\n {%- endfor %}\n {{- '\n\n\n<tools>' }}\n {{- '\n\n\nFor each function call, output the function name and arguments in the following format:\n\n\n<tool_call>\n\n<function=example_function_name>\n\n<parameter=example_parameter_1>value_1</parameter>\n\n<parameter=example_parameter_2>This is the value for the second parameter\n\nthat can span\n\nmultiple  
lines</parameter>\n\n</function>\n\n\n</tool_call>\n\n\n<IMPORTANT>\n\n- Function calls MUST follow the specified  
format: an inner <function=...></function> block must be nested within <tool_call></tool_call> XML tags\n- DO  
NOT use function calls inside <think></think> tags.\n- The value enclosed between parameter tags is preserved  
exactly as-is, including newlines and spaces.\n\n\n</IMPORTANT>' }}\n {%- endif %}\n {{- '<|im_end|>' }}\n {%- for  
message in loop_messages %}\n {%- if message.content is string %}\n {%- set content = message.content %}\n {%-  
else %}\n {%- set content = render_content(message.content) %}\n {%- endif %}\n {%- if message.role ==  
\"assistant\" %}\n {%- if message.reasoning_content is string %}\n {%- set reasoning_content =  
message.reasoning_content %}\n {%- else %}\n {%- set reasoning_content = \"\" %}\n {%- if '</think>' in content  
%}\n {%- set reasoning_content = content.split('</think>')[0].split('<think>')[-1] %}\n {%- set content =  
content.split('</think>')[-1] %}\n {%- endif %}\n {%- endif %}\n {%- if (keep_all_reasoning or loop.index0 >  
ns.last_user_index) and reasoning_content -%}\n {{- '<|im_start|>' + message.role + '\n\n<think>' +  
reasoning_content + '</think>' + content }}\n {%- else %}\n {{- '<|im_start|>' + message.role + '\n\n<think></think>' +  
content }}\n {%- endif %}\n {%- if message.tool_calls is defined and message.tool_calls is iterable and
```

```

message.tool_calls | length > 0 %}\n {%- for tool_call in message.tool_calls %}\n {%- if tool_call.function is defined
%}\n {%- set tool_call = tool_call.function %}\n {%- endif %}\n {{- '<tool_call>\n<function=' + tool_call.name + '>\n'
}}\n {%- if tool_call.arguments is defined %}\n {%- for args_name, args_value in tool_call.arguments.items %}\n {{-
'<parameter=' + args_name + '>' }}\n {%- set args_value = args_value | tojson | safe if args_value is mapping or
(args_value is sequence and args_value is not string) else args_value | string %}\n {{- args_value }}\n {{- '</
parameter>\n' }}\n {%- endfor %}\n {%- endif %}\n {{- '</function>\n</tool_call>' }}\n {%- endfor %}\n {%- endif
%}\n {{- '<|im_end|>' }}\n {%- elif message.role == \"user\" %}\n {{- '<|im_start|>' + message.role + '\n' +
render_content(message.content) + '<|im_end|>' }}\n {%- elif message.role == \"system\" %}\n {{- '<|im_start|>' +
message.role + '\n' + render_content(message.content) + '<|im_end|>' }}\n {%- elif message.role == \"tool\" %}\n
{%- if loop.previtem and loop.previtem.role != \"tool\" %}\n {{- '<|im_start|>tool\n' }}\n {%- endif %}\n {{-
'<tool_response>\n' }}\n {{- render_content(message.content) }}\n {{- '\n</tool_response>\n' }}\n {%- if not
loop.last and loop.nextitem.role != \"tool\" %}\n {{- '<|im_end|>' }}\n {%- elif loop.last %}\n {{- '<|im_end|>' }}\n {%-
endif %}\n {%- else %}\n {{- '<|im_start|>' + message.role + '\n' + render_content(message.content) + '<|im_end|>'
}}\n {%- endif %}\n {%- endfor %}\n {%- if add_generation_prompt %}\n {{- '<|im_start|>assistant\n' }}\n {%- if not
enable_thinking -%}\n {{- '<think></think>' -}}\n {%- else -%}\n {{- \" \" -}}\n {%- endif -%}\n {%- endif
%}\n\n\"eos_token\": \"<|im_end|>\", \"pad_token\": \"</endofextl
>\", \"unk_token\": null}}, \"createdAt\": \"2026-04-27T13:37:38.000Z\", \"discussionsDisabled\": false, \"discussionsSorting\": \"recent
created\", \"downloads\": 170204, \"downloadsAllTime\": 1583227, \"id\": \"XiaomiMiMo/MiMo-
V2.5\", \"isLikedByUser\": false, \"availableInferenceProviders\":
[{\"provider\": \"novita\", \"modelStatus\": \"live\", \"providerStatus\": \"live\", \"providerId\": \"xiaomimimo/mimo-
v2.5\", \"task\": \"conversational\", \"adapterWeightsPath\": \"audio_tokenizer/model.safetensors\", \"features\":
{\"structuredOutput\": false, \"toolCalling\": true}, \"isCheapestPricingOutput\": false, \"isFastestThroughput\": true, \"isModelAuth
{\"provider\": \"featherless-ai\", \"modelStatus\": \"live\", \"providerStatus\": \"live\", \"providerId\": \"XiaomiMiMo/MiMo-
V2.5\", \"task\": \"conversational\", \"adapterWeightsPath\": \"audio_tokenizer/
model.safetensors\", \"isCheapestPricingOutput\": false, \"isFastestThroughput\": false, \"isModelAuthor\": false},
{\"provider\": \"deepinfra\", \"modelStatus\": \"live\", \"providerStatus\": \"live\", \"providerId\": \"XiaomiMiMo/MiMo-
V2.5\", \"task\": \"conversational\", \"adapterWeightsPath\": \"audio_tokenizer/model.safetensors\", \"features\":
{\"structuredOutput\": true, \"toolCalling\": true}, \"isCheapestPricingOutput\": true, \"isFastestThroughput\": false, \"isModelAuthc
generation\", \"library_name\": \"transformers\", \"librariesOther\": [], \"trackDownloads\": true, \"model-
index\": null, \"evalResults\": [{\"dataset\": {\"id\": \"ScaleAI/SWE-
bench_Pro\", \"isBenchmark\": true, \"task_id\": \"SWE_Bench_Pro\", \"value\": 56.1, \"source\": {\"url\": \"https://huggingface.co/
XiaomiMiMo/MiMo-V2.5\", \"name\": \"Model Card\", \"isExternal\": false}, \"filename\": \"eval_results/swe-
bench_pro.yaml\", \"verified\": false, \"pullRequest\": 1, \"rank\": 24, \"label\": \"SWE Bench Pro\", {\"dataset\":
{\"id\": \"harborframework/terminal-bench-2.0\", \"isBenchmark\": true, \"task_id\": \"terminalbench_2\", \"value\": 65.8, \"source\":
{\"url\": \"https://huggingface.co/XiaomiMiMo/MiMo-V2.5\", \"name\": \"Model
Card\", \"isExternal\": false}, \"filename\": \"eval_results/terminal-
bench-2.0.yaml\", \"verified\": false, \"pullRequest\": 1, \"rank\": 8, \"sizeRank\":
{\"rank\": 1, \"value\": 500000000000, \"id\": \"500B\", \"label\": \"< 500B\", \"label\": \"Terminalbench 2\", {\"dataset\":
{\"id\": \"InternScience/ResearchClawBench\", \"isBenchmark\": false, \"task_id\": \"overall\", \"value\": 16.91, \"source\":
{\"url\": \"https://huggingface.co/XiaomiMiMo/MiMo-V2.5\", \"name\": \"Model
Card\", \"isExternal\": false}, \"filename\": \"eval_results/
researchclawbench.yaml\", \"verified\": false, \"pullRequest\": 17, \"rank\": 5, \"sizeRank\":
{\"rank\": 1, \"value\": 500000000000, \"id\": \"500B\", \"label\": \"< 500B\", \"label\": \"Overall\", \"notes\": \"ResearchHarness
evaluation with tools enabled, code execution, and a file-system workspace; completed 39/40
ResearchClawBench tasks.\"}, {\"dataset\": {\"id\": \"claw-eval/Claw-
Eval\", \"isBenchmark\": true, \"task_id\": \"general\", \"value\": 62.1, \"source\": {\"url\": \"https://claw-
eval.github.io\", \"name\": \"Claw-Eval Leaderboard\", \"isExternal\": true, \"author\":
{\"_id\": \"6038d6d0612f5eef3cc05ea9\", \"avatarUrl\": \"https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg\", \"fullname\": \"Lei
Li\", \"name\": \"tobiaslee\", \"type\": \"user\", \"isPro\": false, \"isHf\": false, \"isHfAdmin\": false, \"isMod\": false, \"followerCount\": 30, \"isUse
claw_eval.yaml\", \"verified\": false, \"pullRequest\": 10, \"rank\": 14, \"label\": \"General\", \"notes\": \"Pass% | N=3 | 161 tasks\",
{\"dataset\": {\"id\": \"claw-eval/Claw-Eval\", \"isBenchmark\": true, \"task_id\": \"multimodal\", \"value\": 23.8, \"source\":
{\"url\": \"https://claw-eval.github.io\", \"name\": \"Claw-Eval Leaderboard\", \"isExternal\": true, \"author\":
{\"_id\": \"6038d6d0612f5eef3cc05ea9\", \"avatarUrl\": \"https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg\", \"fullname\": \"Lei
Li\", \"name\": \"tobiaslee\", \"type\": \"user\", \"isPro\": false, \"isHf\": false, \"isHfAdmin\": false, \"isMod\": false, \"followerCount\": 30, \"isUse

```

```

claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 3, "sizeRank":
{"rank": 3, "value": 50000000000, "id": "500B", "label": "< 500B", "label": "Multimodal", "notes": "Pass3% | N=3 | 101
tasks"}, {"dataset": {"id": "claw-eval/Claw-Eval", "isBenchmark": true, "task_id": "multi_turn"}, "value": 63.2, "source":
{"url": "https://claw-eval.github.io", "name": "Claw-Eval Leaderboard", "isExternal": true, "author":
{"_id": "6038d6d0612f5eef3cc05ea9", "avatarUrl": "https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg", "fullname": "Lei
Li", "name": "tobiaslee", "type": "user", "isPro": false, "isHf": false, "isHfAdmin": false, "isMod": false, "followerCount": 30, "isUse
claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 3, "sizeRank":
{"rank": 1, "value": 50000000000, "id": "500B", "label": "< 500B", "label": "Multi Turn", "notes": "Pass3% | N=3 | 38
tasks"}, {"private": false, "repoType": "model", "gated": false, "tags": ["transformers", "safetensors", "mimo_v2", "text-
generation", "multimodal", "vision-language", "audio", "agent", "video-understanding", "long-
context", "conversational", "custom_code", "en", "zh", "license:mit", "eval-results", "fp8", "region:us"], "tag_objs":
[{"id": "text-generation", "label": "Text Generation", "type": "pipeline_tag", "subType": "nlp"},
{"id": "transformers", "label": "Transformers", "type": "library"}, {"id": "safetensors", "label": "Safetensors", "type": "library"},
{"id": "en", "label": "English", "type": "language"}, {"id": "zh", "label": "Chinese", "type": "language"},
{"id": "mimo_v2", "label": "mimo_v2", "type": "other", "clickable": true},
{"id": "multimodal", "label": "multimodal", "type": "other", "clickable": true}, {"id": "vision-language", "label": "vision-
language", "type": "other", "clickable": true}, {"id": "audio", "label": "audio", "type": "other", "clickable": true},
{"id": "agent", "label": "agent", "type": "other", "clickable": true}, {"id": "video-understanding", "label": "video-
understanding", "type": "other", "clickable": true}, {"id": "long-context", "label": "long-
context", "type": "other", "clickable": true},
{"id": "conversational", "label": "conversational", "type": "other", "clickable": true},
{"id": "custom_code", "label": "custom_code", "type": "other", "clickable": true}, {"id": "eval-results", "label": "Eval
Results", "type": "other", "clickable": true}, {"id": "fp8", "label": "fp8", "type": "other", "clickable": true},
{"id": "license:mit", "label": "mit", "type": "license"}, {"type": "region", "label": "🌐 Region:
US", "id": "region:us"}], "transformersInfo":
{"auto_model": "AutoModelForCausalLM", "custom_class": "modeling_mimo_v2.MiMoV2ForCausalLM", "pipeline_tag": "
generation"}, {"widgetData": [{"text": "Hi, what can you help me with?"}, {"text": "What is 84 * 3 / 2?"}, {"text": "Tell me an
interesting fact about the universe!"}, {"text": "Explain quantum computing in simple terms."}], "safetensors":
{"parameters":
{"F32": 68012416, "BF16": 4052024960, "F8_E4M3": 306655002624}, {"total": 310775040000, "sharded": true, "totalFileSize":
{"closed": 4, "open": 20, "total": 24}, {"query": {}, {"inferenceContextData": {"billableEntities": [], "entityName2Providers":
{}}, {"defaultProviders":
[{"isOriginalProvider": false, "name": "novita", "enabled": true, "position": 1, "isReleased": true, "accuratePricing": true},
{"isOriginalProvider": false, "name": "deepinfra", "enabled": true, "position": 16, "isReleased": true, "accuratePricing": true},
{"isOriginalProvider": false, "name": "featherless-
ai", "enabled": true, "position": 7, "isReleased": true, "accuratePricing": true}]]}, {"hasQuantizations": true, "copyToBucketNam
[]}]>

```

| [HuggingFace](#) | [📖 Blog](#) | [🤖 Xiaomi MiMo API Platform](#) | [🗨️ Xiaomi MiMo Studio](#) |

Community

[WeChat Group](#) | [Discord](#) | [Telegram](#) | [Reddit](#)

⚠️ Important: Config Update Notice

The config.json and tokenizer_config.json files in this repository have been updated since the initial release. If you downloaded MiMo-V2.5 before this commit (4da2748) , please re-pull or manually update these two files to ensure correct model behavior. Using the outdated config may lead to degraded model performance. We apologize for any inconvenience.

Quick fix: hf download XiaomiMiMo/MiMo-V2.5 config.json tokenizer_config.json --local-dir ./MiMo-V2.5

MiMo-V2.5

1. Introduction

MiMo-V2.5 is a native omnimodal model with strong agentic capabilities, supporting text, image, video, and audio understanding within a unified architecture. Built upon the MiMo-V2-Flash backbone and extended with dedicated

vision and audio encoders, it delivers robust performance across multimodal perception, long-context reasoning, and agentic workflows. Key features include:

Hybrid Attention Architecture : Inherits the hybrid design from MiMo-V2-Flash, interleaving Sliding Window Attention (SWA) and Global Attention (GA) with a 5:1 ratio and 128 sliding window. This reduces KV-cache storage by nearly 6× while maintaining long-context performance via learnable attention sink bias.

Native Omnimodal Encoders : Equipped with a 729M-param Vision Transformer (ViT) featuring hybrid window attention and a dedicated audio encoder initialized from the weights of MiMo-Audio, enabling high-quality image, video, and audio understanding.

Multi-Token Prediction (MTP) : Three lightweight MTP modules with dense FFNs accelerate inference via speculative decoding and improve RL training efficiency.

Efficient Pre-Training : Trained on a total of ~48T tokens using FP8 mixed precision. The context window supports up to 1M tokens.

Agentic Capabilities : Post-training incorporates SFT, large-scale agentic RL, and Multi-Teacher On-Policy Distillation (MOPD), achieving strong performance on agentic tasks and multimodal understanding benchmarks.

Model Summary

Architecture : Sparse MoE (Mixture of Experts), 310B total / 15B activated parameters

Context Length : Up to 1M tokens

Modalities : Text, Image, Video, Audio

Vision Encoder : 729M-param ViT (28 layers: 24 SWA + 4 Full)

Audio Encoder : 261M-param Audio Transformer (24 layers: 12 SWA + 12 Full)

Multi-Token Prediction (MTP) : 329M parameters, 3 layers

2. Downloads

Model Context Length Download

MiMo-V2.5-Base 256K [HuggingFace](#) [ModelScope](#)

MiMo-V2.5 1M [HuggingFace](#) [ModelScope](#)

3. Evaluation Results

Multimodal Benchmarks

Coding & Agent Benchmarks

Long Context Benchmarks

4. Model Architecture

LLM Backbone

MiMo-V2.5's core language backbone inherits from the MiMo-V2-Flash architecture, a sparse MoE model with hybrid sliding window attention.

Component MiMo-V2.5-Pro MiMo-V2.5

Total Parameters 1.02T 310B

Activated Parameters 42B 15B

Hidden Size 6144 4096

Num Layers 70 (1 dense + 69 MoE) 48 (1 dense + 47 MoE)

Full Attention Layers 10 9

SWA Layers 60 39

Num Attention Heads 128 64

Num KV Heads 8 (GQA) 8 (GA) / 4 (SWA)

Head Dim (QK / V) 192 / 128 192 / 128

Routed Experts 384 256

Experts per Token 8 8

MoE Intermediate Size 2048 2048

Dense Intermediate Size 16384 (layer 0 only) 16384 (layer 0 only)

SWA Window Size 128 128

Max Context Length 1M 1M

MTP Layers 3 3

Vision Encoder

We train a dedicated MiMo ViT that adopts sliding-window attention to enable efficient visual encoding.

Configuration Value

Total Layers 28

SWA Layers 24

Full Attention Layers 4

Window-Attention Pattern [-1] + [0,0,0,0,1,1,1,1,-1] × 3

Attention Heads (Q / KV) 32 / 8

Head Dimensions (QK / V) 64 / 64

Sliding Window Size (L / R) 64 / 64

Window pattern notation: -1 = full attention, 0 = 1-D row window, 1 = 1-D column window.

Audio Encoder

Our audio encoder is initialized from the weights of MiMo-Audio-Tokenizer and further finetuned to support high-quality audio understanding.

Configuration Value

Total Layers 24

SWA Layers 12

Full Attention Layers 12

Sliding Window Size 128

Attention Heads (Q / KV) 16 / 16

Head Dimensions (QK / V) 64 / 64

5. Training Process

MiMo-V2.5 is trained on a total of ~48T tokens.

Text Pre-training : We collect diverse text data for pre-training the LLM backbone.

Projector Warmup : Short-duration warmup of multimodal projectors (audio and visual MLP projectors).

Multimodal Pre-training : High-quality multimodal data collected for large-scale pretraining.

SFT & Agentic Post Training : Supervised fine-tuning with diverse agentic data. During this stage, the context

window is progressively extended from 32K → 256K → 1M.

RL & MOPD Training : Reinforcement learning for improving perception, reasoning, and agentic capabilities.

6. Deployment

Since inference engines are continuously being updated and optimized, this guide only provides deployment examples for reference. For the best performance, we strongly recommend following our referenced approach to get the latest best practices and optimal performance.

SGLang Deployment

For the best performance, we strongly recommend deploying using this approach, which is officially supported by the SGLang community. Please refer to SGLang MiMo-V2.5 Cookbook for the latest deployment guide.

The following is an example of running the model with SGLang, referenced from [sgl-project/sglang#23811](https://github.com/sgl-project/sglang/pull/23811) :

```
python3 -m sglang.launch_server \
--model-path XiaomiMiMo/MiMo-V2.5 \
--served-model-name mimo-v2.5 \
--log-level-http warning \
--enable-cache-report \
--pp-size 1 \
--dp-size 2 \
--tp-size 8 \
--enable-dp-attention \
--moe-a2a-backend deeppep \
--deeppep-mode auto \
--decode-log-interval 1 \
--page-size 1 \
--host 0.0.0.0 \
--port 9001 \
--trust-remote-code \
--watchdog-timeout 1000000 \
--mem-fraction-static 0.65 \
--chunked-prefill-size 16384 \
--reasoning-parser qwen3 \
--tool-call-parser mimo \
--context-length 262144 \
--collect-tokens-histogram \
--enable-metrics \
--load-balance-method round_robin \
--allow-auto-truncate \
--enable-metrics-for-all-schedulers \
--quantization fp8 \
--skip-server-warmup \
--moe-dense-tp-size 1 \
--enable-dp-lm-head \
--disable-tokenizer-batch-decode \
--mm-enable-dp-encoder \
--attention-backend fa3 \
--mm-attention-backend fa3
```

vLLM Deployment

For the best performance, we strongly recommend deploying using this approach, which is officially supported by the vLLM community. Please refer to vLLM MiMo-V2-Flash Cookbook for the latest deployment guide.

For local deployment, we recommend setting the sampling parameters to temperature=1.0 , top_p=0.95 .

Citation

@misc{mimov25, title={MiMo-V2.5}, year={2026}, howpublished={\url{https://huggingface.co/collections/XiaomiMiMo/mimo-v25}}, }

Contact

For questions or feedback, reach us at mimo@xiaomi.com or join our community:

WeChat Group

Discord

Telegram

Reddit

Downloads last month 170,204

```
' ~ (json_dict[json_key] | tojson | safe) ~ '<' ~ json_key ~ '>' }}\n {%- else %}\n {{-'\n<' ~ json_key ~ '>' ~\n(json_dict[json_key] | string) ~ '</' ~ json_key ~ '>' }}\n {%- endif %}\n {%- endfor %}\n {%- endif %}\n {%- endmacro\n-%}\n{%- macro render_content(message_content) -%}\n {%- if message_content is string -%}\n {{-\nmessage_content -}}\n {%- else -%}\n {%- for content in message_content -%}\n {%- if content['type'] == 'image' or\n'image' in content or 'image_url' in content -%}\n {{- '<|vision_start|><|image_pad|><|vision_end|>' -}}\n {%- elif\ncontent['type'] == 'audio' or 'audio' in content or 'audio_url' in content -%}\n {{- '<|mimo_audio_start|><|audio_pad|\n><|mimo_audio_end|>' -}}\n {%- elif content['type'] == 'video' or 'video' in content or 'video_url' in content -%}\n {{-\n'<|vision_start|><|video_pad|><|vision_end|>' -}}\n {%- elif 'text' in content -%}\n {{- content['text'] -}}\n {%- endif -\n%}\n {%- endfor -%}\n {%- endif -%}\n {%- endmacro -%}\n {%- if messages[0]['role'] == 'system' %}\n {%- set\nsystem_message = messages[0]['content'] %}\n {%- set loop_messages = messages[1:] %}\n {%- else %}\n {%-\nset loop_messages = messages %}\n {%- endif %}\n {%- set ns = namespace(last_user_index=-1) %}\n {%- for m in\nloop_messages %}\n {%- if m.role == 'user' %}\n {%- set ns.last_user_index = loop.index0 -%}\n {%- endif\n%}\n {%- endfor %}\n {%- if not tools is defined %}\n {%- set tools = [] %}\n {%- endif %}\n {%- if system_message is
```

```

defined %}\n {{- '<|im_start|>system\\n\\n' + render_content(system_message) }}\n%- else %}\n {{- '<|im_start|
>system\\n\\nYou are MiMo, a helpful AI assistant engineered by Xiaomi.' }}\n%- endif %}\n%- if tools is iterable
and tools | length > 0 %}\n {{- '\\n\\n# Tools\\n\\nYou may call one or more functions to assist with the user query.\\
\\n\\nYou have access to the following functions:\\n\\n' }}\n {{- '<tools>' }}\n %- for tool in tools %}\n %- if
tool.function is defined %}\n %- set tool = tool.function %}\n %- endif %}\n {{- '\\n<function>\\n<name>' ~
tool.name ~ '</name>' }}\n %- if tool.description is defined %}\n {{- '\\n<description>' ~ (tool.description | trim) ~
'</description>' }}\n %- endif %}\n {{- '\\n<parameters>' }}\n %- if tool.parameters is defined and tool.parameters
is mapping and tool.parameters.properties is defined and tool.parameters.properties is mapping %}\n %- for
param_name, param_fields in tool.parameters.properties.items %}\n {{- '\\n<parameter>' }}\n {{- '\\n<name>' ~
param_name ~ '</name>' }}\n %- if param_fields.type is defined %}\n {{- '\\n<type>' ~ (param_fields.type | string)
~ '</type>' }}\n %- endif %}\n %- if param_fields.description is defined %}\n {{- '\\n<description>' ~
(param_fields.description | trim) ~ '</description>' }}\n %- endif %}\n %- set handled_keys = ['name', 'type',
'description'] %}\n {{- render_extra_keys(param_fields, handled_keys) }}\n {{- '\\n</parameter>' }}\n %- endfor
%}\n %- endif %}\n %- set handled_keys = ['type', 'properties'] %}\n {{- render_extra_keys(tool.parameters,
handled_keys) }}\n {{- '\\n</parameters>' }}\n %- set handled_keys = ['type', 'name', 'description', 'parameters']
%}\n {{- render_extra_keys(tool, handled_keys) }}\n {{- '\\n</function>' }}\n %- endfor %}\n {{- '\\n</tools>' }}\n {{-
'\\n\\nFor each function call, output the function name and arguments in the following format:\\n<tool_call>\\
\\n<function=example_function_name>\\n<parameter=example_parameter_1>value_1</parameter>\\
\\n<parameter=example_parameter_2>This is the value for the second parameter\\n\\nthat can span\\n\\nmultiple
lines</parameter>\\n</function>\\n</tool_call>\\n\\n<IMPORTANT>\\n- Function calls MUST follow the specified
format: an inner <function=...></function> block must be nested within <tool_call></tool_call> XML tags\\n- DO
NOT use function calls inside <think></think> tags.\\n- The value enclosed between parameter tags is preserved
exactly as-is, including newlines and spaces.\\n</IMPORTANT>' }}\n %- endif %}\n {{- '<|im_end|>' }}\n %- for
message in loop_messages %}\n %- if message.content is string %}\n %- set content = message.content %}\n
%- else %}\n %- set content = render_content(message.content) %}\n %- endif %}\n %- if message.role ==
'assistant' %}\n %- if message.reasoning_content is string %}\n %- set reasoning_content =
message.reasoning_content %}\n %- else %}\n %- set reasoning_content = " %}\n %- if '</think>' in content
%}\n %- set reasoning_content = content.split('</think>')[0].split('<think>')[-1] %}\n %- set content =
content.split('</think>')[-1] %}\n %- endif %}\n %- endif %}\n %- if (keep_all_reasoning or loop.index0 >
ns.last_user_index) and reasoning_content != '' %}\n {{- '<|im_start|>' + message.role + '\\n<think>' +
reasoning_content + '</think>' + content }}\n %- else %}\n {{- '<|im_start|>' + message.role + '\\n<think></think>' +
content }}\n %- endif %}\n %- if message.tool_calls is defined and message.tool_calls is iterable and
message.tool_calls | length > 0 %}\n %- for tool_call in message.tool_calls %}\n %- if tool_call.function is defined
%}\n %- set tool_call = tool_call.function %}\n %- endif %}\n {{- '<tool_call>\\n<function=' + tool_call.name + '>\\n'
}}\n %- if tool_call.arguments is defined %}\n %- for args_name, args_value in tool_call.arguments.items %}\n {{-
'<parameter=' + args_name + '>' }}\n %- set args_value = args_value | tojson | safe if args_value is mapping or
(args_value is sequence and args_value is not string) else args_value | string %}\n {{- args_value }}\n {{- '</
parameter>\\n' }}\n %- endfor %}\n %- endif %}\n {{- '</function>\\n</tool_call>' }}\n %- endfor %}\n %- endif
%}\n {{- '<|im_end|>' }}\n %- elif message.role == 'user' %}\n {{- '<|im_start|>' + message.role + '\\n' +
render_content(message.content) + '<|im_end|>' }}\n %- elif message.role == 'system' %}\n {{- '<|im_start|>' +
message.role + '\\n' + render_content(message.content) + '<|im_end|>' }}\n %- elif message.role == 'tool' %}\n
%- if loop.previtem and loop.previtem.role != 'tool' %}\n {{- '<|im_start|>tool\\n' }}\n %- endif %}\n {{-
'<tool_response>\\n' }}\n {{- render_content(message.content) }}\n {{- '\\n</tool_response>\\n' }}\n %- if not
loop.last and loop.nextitem.role != 'tool' %}\n {{- '<|im_end|>' }}\n %- elif loop.last %}\n {{- '<|im_end|>' }}\n %-
endif %}\n %- else %}\n {{- '<|im_start|>' + message.role + '\\n' + render_content(message.content) + '<|im_end|>'
}}\n %- endif %}\n %- endfor %}\n %- if add_generation_prompt %}\n {{- '<|im_start|>assistant\\n' }}\n %- if not
enable_thinking -%}\n {{- '<think></think>' -%}}\n %- else -%}\n {{- " " -%}}\n %- endif -%}\n %- endif
%}\n\\n", "eos_token": "<|im_end|>", "pad_token": "</endoftext|
>", "unk_token": null}}, "createdAt": "2026-04-27T13:37:38.000Z", "discussionsDisabled": false, "discussionsSorting": "recently
created", "downloads": 170204, "downloadsAllTime": 1583227, "id": "XiaomiMiMo/MiMo-
V2.5", "isLikedByUser": false, "availableInferenceProviders":
[{"provider": "novita", "modelStatus": "live", "providerStatus": "live", "providerId": "xiaomimimo/mimo-
v2.5", "task": "conversational", "adapterWeightsPath": "audio_tokenizer/model.safetensors", "features":
{"structuredOutput": false, "toolCalling": true}, "isCheapestPricingOutput": false, "isFastestThroughput": true, "isModelAuth":
{"provider": "featherless-ai", "modelStatus": "live", "providerStatus": "live", "providerId": "XiaomiMiMo/MiMo-
V2.5", "task": "conversational", "adapterWeightsPath": "audio_tokenizer/

```

```

model.safetensors", "isCheapestPricingOutput": false, "isFastestThroughput": false, "isModelAuthor": false,
{"provider": "deepinfra", "modelStatus": "live", "providerStatus": "live", "providerId": "XiaomiMiMo/MiMo-
V2.5", "task": "conversational", "adapterWeightsPath": "audio_tokenizer/model.safetensors", "features":
{"structuredOutput": true, "toolCalling": true}, "isCheapestPricingOutput": true, "isFastestThroughput": false, "isModelAuthc
generation", "library_name": "transformers", "librariesOther": [], "trackDownloads": true, "model-
index": null, "evalResults": [{"dataset": {"id": "ScaleAI/SWE-
bench_Pro", "isBenchmark": true, "task_id": "SWE_Bench_Pro", "value": 56.1, "source": {"url": "https://huggingface.co/
XiaomiMiMo/MiMo-V2.5", "name": "Model Card", "isExternal": false}, "filename": ".eval_results/swe-
bench_pro.yaml", "verified": false, "pullRequest": 1, "rank": 24, "label": "SWE Bench Pro"}, {"dataset":
{"id": "harborframework/terminal-bench-2.0", "isBenchmark": true, "task_id": "terminalbench_2", "value": 65.8, "source":
{"url": "https://huggingface.co/XiaomiMiMo/MiMo-V2.5", "name": "Model
Card", "isExternal": false}, "filename": ".eval_results/terminal-
bench-2.0.yaml", "verified": false, "pullRequest": 1, "rank": 8, "sizeRank":
{"rank": 1, "value": 500000000000, "id": "500B", "label": "< 500B"}, {"label": "Terminalbench 2"}, {"dataset":
{"id": "InternScience/ResearchClawBench", "isBenchmark": false, "task_id": "overall", "value": 16.91, "source":
{"url": "https://huggingface.co/XiaomiMiMo/MiMo-V2.5", "name": "Model
Card", "isExternal": false}, "filename": ".eval_results/
researchclawbench.yaml", "verified": false, "pullRequest": 17, "rank": 5, "sizeRank":
{"rank": 1, "value": 500000000000, "id": "500B", "label": "< 500B"}, {"label": "Overall", "notes": "ResearchHarness
evaluation with tools enabled, code execution, and a file-system workspace; completed 39/40
ResearchClawBench tasks."}, {"dataset": {"id": "claw-eval/Claw-
Eval", "isBenchmark": true, "task_id": "general", "value": 62.1, "source": {"url": "https://claw-
eval.github.io", "name": "Claw-Eval Leaderboard", "isExternal": true, "author":
{"_id": "6038d6d0612f5eef3cc05ea9", "avatarUrl": "https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrIIQZ1f.jpeg", "fullname": "Lei
Li", "name": "tobiaslee", "type": "user", "isPro": false, "isHf": false, "isHfAdmin": false, "isMod": false, "followerCount": 30, "isUse
claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 14, "label": "General", "notes": "Pass3% | N=3 | 161 tasks"},
{"dataset": {"id": "claw-eval/Claw-Eval", "isBenchmark": true, "task_id": "multimodal", "value": 23.8, "source":
{"url": "https://claw-eval.github.io", "name": "Claw-Eval Leaderboard", "isExternal": true, "author":
{"_id": "6038d6d0612f5eef3cc05ea9", "avatarUrl": "https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrIIQZ1f.jpeg", "fullname": "Lei
Li", "name": "tobiaslee", "type": "user", "isPro": false, "isHf": false, "isHfAdmin": false, "isMod": false, "followerCount": 30, "isUse
claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 3, "sizeRank":
{"rank": 3, "value": 500000000000, "id": "500B", "label": "< 500B"}, {"label": "Multimodal", "notes": "Pass3% | N=3 | 101
tasks"}, {"dataset": {"id": "claw-eval/Claw-Eval", "isBenchmark": true, "task_id": "multi_turn", "value": 63.2, "source":
{"url": "https://claw-eval.github.io", "name": "Claw-Eval Leaderboard", "isExternal": true, "author":
{"_id": "6038d6d0612f5eef3cc05ea9", "avatarUrl": "https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrIIQZ1f.jpeg", "fullname": "Lei
Li", "name": "tobiaslee", "type": "user", "isPro": false, "isHf": false, "isHfAdmin": false, "isMod": false, "followerCount": 30, "isUse
claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 3, "sizeRank":
{"rank": 1, "value": 500000000000, "id": "500B", "label": "< 500B"}, {"label": "Multi Turn", "notes": "Pass3% | N=3 | 38
tasks"}}, {"private": false, "repoType": "model", "gated": false, "tags": ["transformers", "safetensors", "mimo_v2", "text-
generation", "multimodal", "vision-language", "audio", "agent", "video-understanding", "long-
context", "conversational", "custom_code", "en", "zh", "license:mit", "eval-results", "fp8", "region:us"], "tag_objs":
[{"id": "text-generation", "label": "Text Generation", "type": "pipeline_tag", "subType": "nlp"},
{"id": "transformers", "label": "Transformers", "type": "library"}, {"id": "safetensors", "label": "Safetensors", "type": "library"},
{"id": "en", "label": "English", "type": "language"}, {"id": "zh", "label": "Chinese", "type": "language"},
{"id": "mimo_v2", "label": "mimo_v2", "type": "other", "clickable": true},
{"id": "multimodal", "label": "multimodal", "type": "other", "clickable": true}, {"id": "vision-language", "label": "vision-
language", "type": "other", "clickable": true}, {"id": "audio", "label": "audio", "type": "other", "clickable": true},
{"id": "agent", "label": "agent", "type": "other", "clickable": true}, {"id": "video-understanding", "label": "video-
understanding", "type": "other", "clickable": true}, {"id": "long-context", "label": "long-
context", "type": "other", "clickable": true},
{"id": "conversational", "label": "conversational", "type": "other", "clickable": true},
{"id": "custom_code", "label": "custom_code", "type": "other", "clickable": true}, {"id": "eval-results", "label": "Eval
Results", "type": "other", "clickable": true}, {"id": "fp8", "label": "fp8", "type": "other", "clickable": true},

```

```

{"id": "license:mit", "label": "mit", "type": "license"}, {"type": "region", "label": "🇺🇸 Region: US", "id": "region:us"}], "transformersInfo": {"auto_model": "AutoModelForCausalLM", "custom_class": "modeling_mimo_v2.MiMoV2ForCausalLM", "pipeline_tag": "generation"}, "widgetData": [{"text": "Hi, what can you help me with?"}, {"text": "What is 84 * 3 / 2?"}, {"text": "Tell me an interesting fact about the universe!"}, {"text": "Explain quantum computing in simple terms."}], "safetensors": {"parameters": {"F32": 68012416, "BF16": 4052024960, "F8_E4M3": 306655002624, "total": 310775040000, "sharded": true, "totalFileSize": 311}, {"billableEntities": [], "entityName2Providers": {}, "defaultProviders": [{"isOriginalProvider": false, "name": "novita", "enabled": true, "position": 1, "isReleased": true, "accuratePricing": true}, {"isOriginalProvider": false, "name": "deepinfra", "enabled": true, "position": 16, "isReleased": true, "accuratePricing": true}, {"isOriginalProvider": false, "name": "featherless-ai", "enabled": true, "position": 7, "isReleased": true, "accuratePricing": true}]]}, {"canWrite": false}]> Safetensors Model size 311B params

```

Tensor type F32 · BF16 · F8_E4M3 ·

Chat template Files info

```
' ~ (json_dict[json_key] | tojson | safe) ~ '<' ~ json_key ~ '>'}}\n{%%- else %}\n{{- '\n<' ~ json_key ~ '>' ~  
(json_dict[json_key] | string) ~ '<' ~ json_key ~ '>'}}\n{%%- endif %}\n{%%- endfor %}\n{%%- endif %}\n{%%- endmacro  
-%}\n{%%- macro render_content(message_content) -%}\n{%%- if message_content is string -%}\n{{-  
message_content -}}\n{%%- else -%}\n{%%- for content in message_content -%}\n{%%- if content['type'] == 'image' or  
'image' in content or 'image_url' in content -%}\n{{- '<|vision_start|><|image_pad|><|vision_end|>' -}}\n{%%- elif  
content['type'] == 'audio' or 'audio' in content or 'audio_url' in content -%}\n{{- '<|mimo_audio_start|><|audio_pad|  
><|mimo_audio_end|>' -}}\n{%%- elif content['type'] == 'video' or 'video' in content or 'video_url' in content -%}\n{{-  
'<|vision_start|><|video_pad|><|vision_end|>' -}}\n{%%- elif 'text' in content -%}\n{{- content['text'] -}}\n{%%- endif -  
%}\n{%%- endfor -%}\n{%%- endif -%}\n{%%- endmacro -%}\n{%%- if messages[0][\"role\"] == \"system\" %}\n{%%- set  
system_message = messages[0][\"content\"] %}\n{%%- set loop_messages = messages[1:] %}\n{%%- else %}\n{%%-  
set loop_messages = messages %}\n{%%- endif %}\n{%%- set ns = namespace(last_user_index=-1) %}\n{%%- for m in  
loop_messages %}\n{%%- if m.role == 'user' %}\n{%%- set ns.last_user_index = loop.index0 -%}\n{%%- endif  
%}\n{%%- endfor %}\n{%%- if not tools is defined %}\n{%%- set tools = [] %}\n{%%- endif %}\n{%%- if system_message is  
defined %}\n{{- '\n<|im_start|>system\n\n' + render_content(system_message) }}\n{%%- else %}\n{{- '\n<|im_start|  
>system\n\nYou are MiMo, a helpful AI assistant engineered by Xiaomi.\n' }}\n{%%- endif %}\n{%%- if tools is iterable  
and tools | length > 0 %}\n{{- '\n\n\n# Tools\n\n\nYou may call one or more functions to assist with the user query.\n\n\nYou have access to the following functions:\n\n\n' }}\n{{- '\n<tools>\n' }}\n{%%- for tool in tools %}\n{%%- if  
tool.function is defined %}\n{%%- set tool = tool.function %}\n{%%- endif %}\n{{- '\n\n<function>\n\n<name>\n' ~  
tool.name ~ '\n</name>\n' }}\n{%%- if tool.description is defined %}\n{{- '\n\n<description>\n' ~ (tool.description | trim) ~  
'</description>\n' }}\n{%%- endif %}\n{{- '\n\n<parameters>\n' }}\n{%%- if tool.parameters is defined and tool.parameters  
is mapping and tool.parameters.properties is defined and tool.parameters.properties is mapping %}\n{%%- for  
param_name, param_fields in tool.parameters.properties.items %}\n{{- '\n\n<parameter>\n' }}\n{{- '\n\n<name>\n' ~  
param_name ~ '\n</name>\n' }}\n{%%- if param_fields.type is defined %}\n{{- '\n\n<type>\n' ~ (param_fields.type | string)  
~ '</type>\n' }}\n{%%- endif %}\n{%%- if param_fields.description is defined %}\n{{- '\n\n<description>\n' ~  
(param_fields.description | trim) ~ '</description>\n' }}\n{%%- endif %}\n{%%- set handled_keys = ['name', 'type',  
'description'] %}\n{{- render_extra_keys(param_fields, handled_keys) }}\n{{- '\n\n</parameter>\n' }}\n{%%- endfor  
%}\n{%%- endif %}\n{%%- set handled_keys = ['type', 'properties'] %}\n{%%- render_extra_keys(tool.parameters,  
handled_keys) }}\n{{- '\n\n</parameters>\n' }}\n{%%- set handled_keys = ['type', 'name', 'description', 'parameters']  
%}\n{{- render_extra_keys(tool, handled_keys) }}\n{{- '\n\n</function>\n' }}\n{%%- endfor %}\n{{- '\n\n</tools>\n' }}\n{{-  
'\n\n\nFor each function call, output the function name and arguments in the following format:\n\n\n<tool_call>\n\n<function=example_function_name>\n\n<parameter=example_parameter_1>value_1</parameter>\n\n<parameter=example_parameter_2>This is the value for the second parameter\n\nthat can span\n\nmultiple  
lines</parameter>\n\n</function>\n\n\n</tool_call>\n\n\n<IMPORTANT>\n\n- Function calls MUST follow the specified  
format: an inner <function=...></function> block must be nested within <tool_call></tool_call> XML tags\n\n- DO  
NOT use function calls inside <think></think> tags.\n\n- The value enclosed between parameter tags is preserved  
exactly as-is, including newlines and spaces.\n\n\n</IMPORTANT>\n' }}\n{%%- endif %}\n{{- '<|im_end|>' }}\n{%%- for  
message in loop_messages %}\n{%%- if message.content is string %}\n{%%- set content = message.content %}\n{%%-  
else %}\n{%%- set content = render_content(message.content) %}\n{%%- endif %}\n{%%- if message.role ==  
\"assistant\" %}\n{%%- if message.reasoning_content is string %}\n{%%- set reasoning_content =  
message.reasoning_content %}\n{%%- else %}\n{%%- set reasoning_content = \"\" %}\n{%%- if '</think>' in content
```

```

%}\n {%- set reasoning_content = content.split('</think>')[0].split('<think>')[-1] %}\n {%- set content =
content.split('</think>')[-1] %}\n {%- endif %}\n {%- endif %}\n {%- if (keep_all_reasoning or loop.index0 >
ns.last_user_index) and reasoning_content -%}\n {{- '<|im_start|>' + message.role + '\n<think>' +
reasoning_content + '</think>' + content }}\n {%- else %}\n {{- '<|im_start|>' + message.role + '\n<think></think>' +
content }}\n {%- endif %}\n {%- if message.tool_calls is defined and message.tool_calls is iterable and
message.tool_calls | length > 0 %}\n {%- for tool_call in message.tool_calls %}\n {%- if tool_call.function is defined
%}\n {%- set tool_call = tool_call.function %}\n {%- endif %}\n {{- '<tool_call>\n<function=' + tool_call.name + '>\n'
}}\n {%- if tool_call.arguments is defined %}\n {%- for args_name, args_value in tool_call.arguments.items %}\n {{-
'<parameter=' + args_name + '>' }}\n {%- set args_value = args_value | tojson | safe if args_value is mapping or
(args_value is sequence and args_value is not string) else args_value | string %}\n {{- args_value }}\n {{- '</
parameter>\n' }}\n {%- endfor %}\n {%- endif %}\n {{- '</function>\n</tool_call>' }}\n {%- endfor %}\n {%- endif
%}\n {{- '<|im_end|>' }}\n {%- elif message.role == "user" %}\n {{- '<|im_start|>' + message.role + '\n' +
render_content(message.content) + '<|im_end|>' }}\n {%- elif message.role == "system" %}\n {{- '<|im_start|>' +
message.role + '\n' + render_content(message.content) + '<|im_end|>' }}\n {%- elif message.role == "tool" %}\n
{%- if loop.previtem and loop.previtem.role != "tool" %}\n {{- '<|im_start|>tool\n' }}\n {%- endif %}\n {{-
'<tool_response>\n' }}\n {{- render_content(message.content) }}\n {{- '\n</tool_response>\n' }}\n {%- if not
loop.last and loop.nextitem.role != "tool" %}\n {{- '<|im_end|>' }}\n {%- elif loop.last %}\n {{- '<|im_end|>' }}\n {%-
endif %}\n {%- else %}\n {{- '<|im_start|>' + message.role + '\n' + render_content(message.content) + '<|im_end|>'
}}\n {%- endif %}\n {%- endfor %}\n {%- if add_generation_prompt %}\n {{- '<|im_start|>assistant\n' }}\n {%- if not
enable_thinking -%}\n {{- '<think></think>' -}}\n {%- else -%}\n {{- " " -}}\n {%- endif -%}\n {%- endif
%}\n", "eos_token": "<|im_end|>", "pad_token": "<|endoftext|
>", "unk_token": null}}, "createdAt": "2026-04-27T13:37:38.000Z", "discussionsDisabled": false, "discussionsSorting": "recently
created", "downloads": 170204, "downloadsAllTime": 1583227, "id": "XiaomiMiMo/MiMo-
V2.5", "isLikedByUser": false, "availableInferenceProviders":
[{"provider": "novita", "modelStatus": "live", "providerStatus": "live", "providerId": "xiaomimimo/mimo-
v2.5", "task": "conversational", "adapterWeightsPath": "audio_tokenizer/model.safetensors", "features":
{"structuredOutput": false, "toolCalling": true}, "isCheapestPricingOutput": false, "isFastestThroughput": true, "isModelAuth
{"provider": "featherless-ai", "modelStatus": "live", "providerStatus": "live", "providerId": "XiaomiMiMo/MiMo-
V2.5", "task": "conversational", "adapterWeightsPath": "audio_tokenizer/
model.safetensors", "isCheapestPricingOutput": false, "isFastestThroughput": false, "isModelAuthor": false},
{"provider": "deepinfra", "modelStatus": "live", "providerStatus": "live", "providerId": "XiaomiMiMo/MiMo-
V2.5", "task": "conversational", "adapterWeightsPath": "audio_tokenizer/model.safetensors", "features":
{"structuredOutput": true, "toolCalling": true}, "isCheapestPricingOutput": true, "isFastestThroughput": false, "isModelAuthc
generation", "library_name": "transformers", "librariesOther": [], "trackDownloads": true, "model-
index": null, "evalResults": [{"dataset": {"id": "ScaleAI/SWE-
bench_Pro", "isBenchmark": true, "task_id": "SWE_Bench_Pro"}, "value": 56.1, "source": {"url": "https://huggingface.co/
XiaomiMiMo/MiMo-V2.5", "name": "Model Card", "isExternal": false}, "filename": ".eval_results/swe-
bench_pro.yaml", "verified": false, "pullRequest": 1, "rank": 24, "label": "SWE Bench Pro"}, {"dataset":
{"id": "harborframework/terminal-bench-2.0", "isBenchmark": true, "task_id": "terminalbench_2"}, "value": 65.8, "source":
{"url": "https://huggingface.co/XiaomiMiMo/MiMo-V2.5", "name": "Model
Card", "isExternal": false}, "filename": ".eval_results/terminal-
bench-2.0.yaml", "verified": false, "pullRequest": 1, "rank": 8, "sizeRank":
{"rank": 1, "value": 500000000000, "id": "500B", "label": "< 500B"}, "label": "Terminalbench 2"}, {"dataset":
{"id": "InternScience/ResearchClawBench", "isBenchmark": false, "task_id": "overall"}, "value": 16.91, "source":
{"url": "https://huggingface.co/XiaomiMiMo/MiMo-V2.5", "name": "Model
Card", "isExternal": false}, "filename": ".eval_results/
researchclawbench.yaml", "verified": false, "pullRequest": 17, "rank": 5, "sizeRank":
{"rank": 1, "value": 500000000000, "id": "500B", "label": "< 500B"}, "label": "Overall", "notes": "ResearchHarness
evaluation with tools enabled, code execution, and a file-system workspace; completed 39/40
ResearchClawBench tasks."}, {"dataset": {"id": "claw-eval/Claw-
Eval", "isBenchmark": true, "task_id": "general"}, "value": 62.1, "source": {"url": "https://claw-
eval.github.io", "name": "Claw-Eval Leaderboard", "isExternal": true, "author":
{"_id": "6038d6d0612f5eef3cc05ea9", "avatarUrl": "https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrIIZQ1f.jpeg", "fullname": "Lei
Li", "name": "tobiaslee", "type": "user", "isPro": false, "isHf": false, "isHfAdmin": false, "isMod": false, "followerCount": 30, "isUse
claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 14, "label": "General", "notes": "Pass³% | N=3 | 161 tasks"},

```

```

{"dataset":{"id":"claw-eval/Claw-Eval","isBenchmark":true,"task_id":"multimodal"},"value":23.8,"source":
{"url":"https://claw-eval.github.io","name":"Claw-Eval Leaderboard","isExternal":true,"author":
{"_id":"6038d6d0612f5eef3cc05ea9","avatarUrl":"https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg","fullname":"Lei
Li","name":"tobiaslee","type":"user","isPro":false,"isHf":false,"isHfAdmin":false,"isMod":false,"followerCount":30,"isUse
claw_eval.yaml","verified":false,"pullRequest":10,"rank":3,"sizeRank":
{"rank":3,"value":500000000000,"id":"500B","label":"< 500B"},"label":"Multimodal","notes":"Pass3% | N=3 | 101
tasks"},"dataset":{"id":"claw-eval/Claw-Eval","isBenchmark":true,"task_id":"multi_turn"},"value":63.2,"source":
{"url":"https://claw-eval.github.io","name":"Claw-Eval Leaderboard","isExternal":true,"author":
{"_id":"6038d6d0612f5eef3cc05ea9","avatarUrl":"https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg","fullname":"Lei
Li","name":"tobiaslee","type":"user","isPro":false,"isHf":false,"isHfAdmin":false,"isMod":false,"followerCount":30,"isUse
claw_eval.yaml","verified":false,"pullRequest":10,"rank":3,"sizeRank":
{"rank":1,"value":500000000000,"id":"500B","label":"< 500B"},"label":"Multi Turn","notes":"Pass3% | N=3 | 38
tasks"},"private":false,"repoType":"model","gated":false,"tags":["transformers","safetensors","mimo_v2","text-
generation","multimodal","vision-language","audio","agent","video-understanding","long-
context","conversational","custom_code","en","zh","license:mit","eval-results","fp8","region:us"],"tag_objs":
[{"id":"text-generation","label":"Text Generation","type":"pipeline_tag","subType":"nlp"},
{"id":"transformers","label":"Transformers","type":"library"},{"id":"safetensors","label":"Safetensors","type":"library"},
{"id":"en","label":"English","type":"language"},{"id":"zh","label":"Chinese","type":"language"},
{"id":"mimo_v2","label":"mimo_v2","type":"other","clickable":true},
{"id":"multimodal","label":"multimodal","type":"other","clickable":true},{"id":"vision-language","label":"vision-
language","type":"other","clickable":true},{"id":"audio","label":"audio","type":"other","clickable":true},
{"id":"agent","label":"agent","type":"other","clickable":true},{"id":"video-understanding","label":"video-
understanding","type":"other","clickable":true},{"id":"long-context","label":"long-
context","type":"other","clickable":true},
{"id":"conversational","label":"conversational","type":"other","clickable":true},
{"id":"custom_code","label":"custom_code","type":"other","clickable":true},{"id":"eval-results","label":"Eval
Results","type":"other","clickable":true},{"id":"fp8","label":"fp8","type":"other","clickable":true},
{"id":"license:mit","label":"mit","type":"license"},{"type":"region","label":"🌐 Region:
US","id":"region:us"}],"transformersInfo":
{"auto_model":"AutoModelForCausalLM","custom_class":"modeling_mimo_v2.MiMoV2ForCausalLM","pipeline_tag":
"generation"},"widgetData":[{"text":"Hi, what can you help me with?"}, {"text":"What is 84 * 3 / 2?"}, {"text":"Tell me an
interesting fact about the universe!"}, {"text":"Explain quantum computing in simple terms."}], "safetensors":
{"parameters":
{"F32":68012416,"BF16":4052024960,"F8_E4M3":306655002624},"total":310775040000,"sharded":true,"totalFileSize":
{},"inferenceContextData":{"billableEntities":[],"entityName2Providers":{},"defaultProviders":
[{"isOriginalProvider":false,"name":"novita","enabled":true,"position":1,"isReleased":true,"accuratePricing":true},
{"isOriginalProvider":false,"name":"deepinfra","enabled":true,"position":16,"isReleased":true,"accuratePricing":true},
{"isOriginalProvider":false,"name":"featherless-
ai","enabled":true,"position":7,"isReleased":true,"accuratePricing":true}]],"linkToInferenceDiscussion":{"url":"/
spaces/huggingface/InferenceSupport/
discussions/9543","numReactions":15},"isLoggedIn":false,"canWrite":false,"examples":[{"text":"Hi, what can you
help me with?"}, {"text":"What is 84 * 3 / 2?"}, {"text":"Tell me an interesting fact about the universe!"},
{"text":"Explain quantum computing in simple
terms."}], "initialSelectedProvider":"novita","widgetType":"conversational","isMaximized":false,"isMac":true}}">
Inference Providers NEW Novita

```

Text Generation

Examples

Input a message to start chatting with XiaomiMiMo/MiMo-V2.5 .

Send

View Code Snippets

Compare providers

Model tree for XiaomiMiMo/MiMo-V2.5

Adapters

1 model

Quantizations

29 models

Spaces using XiaomiMiMo/MiMo-V2.5 5

Collection including XiaomiMiMo/MiMo-V2.5 4 items • Updated Apr 27 • 92

```
' ~ (json_dict[json_key] | tojson | safe) ~ '<' ~ json_key ~ '>' }}\n {%- else %}\n {{-'\n<' ~ json_key ~ '>' ~  
(json_dict[json_key] | string) ~ '<' ~ json_key ~ '>' }}\n {%- endif %}\n {%- endfor %}\n {%- endif %}\n{%- endmacro  
-%}\n{%- macro render_content(message_content) -%}\n {%- if message_content is string -%}\n {{-  
message_content -}}\n {%- else -%}\n {%- for content in message_content -%}\n {%- if content['type'] == 'image' or  
'image' in content or 'image_url' in content -%}\n {{- '<|vision_start|><|image_pad|><|vision_end|>' -}}\n {%- elif  
content['type'] == 'audio' or 'audio' in content or 'audio_url' in content -%}\n {{- '<|mimo_audio_start|><|audio_pad|  
><|mimo_audio_end|>' -}}\n {%- elif content['type'] == 'video' or 'video' in content or 'video_url' in content -%}\n {{-  
'<|vision_start|><|video_pad|><|vision_end|>' -}}\n {%- elif 'text' in content -%}\n {{- content['text'] -}}\n {%- endif -  
%}\n {%- endfor -%}\n {%- endif -%}\n{%- endmacro -%}\n{%- if messages[0]['role'] == 'system' %}\n {%- set  
system_message = messages[0]['content'] %}\n {%- set loop_messages = messages[1:] %}\n {%- else %}\n {%-  
set loop_messages = messages %}\n {%- endif %}\n {%- set ns = namespace(last_user_index=-1) %}\n {%- for m in  
loop_messages %}\n {%- if m.role == 'user' %}\n {%- set ns.last_user_index = loop.index0 -%}\n {%- endif  
%}\n {%- endfor %}\n {%- if not tools is defined %}\n {%- set tools = [] %}\n {%- endif %}\n {%- if system_message is  
defined %}\n {{- '\n<|im_start|>system\n\n' + render_content(system_message) }}\n {%- else %}\n {{- '\n<|im_start|  
>system\n\nYou are MiMo, a helpful AI assistant engineered by Xiaomi.\n' }}\n {%- endif %}\n {%- if tools is iterable  
and tools | length > 0 %}\n {{- '\n\n\n# Tools\n\n\nYou may call one or more functions to assist with the user query.\n\n\nYou have access to the following functions:\n\n\n' }}\n {{- '\n<tools>\n' }}\n {%- for tool in tools %}\n {%- if  
tool.function is defined %}\n {%- set tool = tool.function %}\n {%- endif %}\n {{- '\n\n<function>\n\n<name>\n' ~  
tool.name ~ '\n</name>\n' }}\n {%- if tool.description is defined %}\n {{- '\n\n<description>\n' ~ (tool.description | trim) ~  
'</description>\n' }}\n {%- endif %}\n {{- '\n\n<parameters>\n' }}\n {%- if tool.parameters is defined and tool.parameters  
is mapping and tool.parameters.properties is defined and tool.parameters.properties is mapping %}\n {%- for  
param_name, param_fields in tool.parameters.properties.items %}\n {{- '\n\n<parameter>\n' }}\n {{- '\n\n<name>\n' ~  
param_name ~ '</name>\n' }}\n {%- if param_fields.type is defined %}\n {{- '\n\n<type>\n' ~ (param_fields.type | string)  
~ '</type>\n' }}\n {%- endif %}\n {%- if param_fields.description is defined %}\n {{- '\n\n<description>\n' ~  
(param_fields.description | trim) ~ '</description>\n' }}\n {%- endif %}\n {%- set handled_keys = ['name', 'type',  
'description'] %}\n {{- render_extra_keys(param_fields, handled_keys) }}\n {{- '\n\n</parameters>\n' }}\n {%- endfor  
%}\n {%- endif %}\n {%- set handled_keys = ['type', 'properties'] %}\n {{- render_extra_keys(tool.parameters,  
handled_keys) }}\n {{- '\n\n</parameters>\n' }}\n {%- set handled_keys = ['type', 'name', 'description', 'parameters']  
%}\n {{- render_extra_keys(tool, handled_keys) }}\n {{- '\n\n</function>\n' }}\n {%- endfor %}\n {{- '\n\n</tools>\n' }}\n {{-  
'\n\n\nFor each function call, output the function name and arguments in the following format:\n\n<tool_call>\n\n<function=example_function_name>\n\n<parameter=example_parameter_1>value_1</parameter>\n\n<parameter=example_parameter_2>This is the value for the second parameter\n\nthat can span\n\nmultiple  
lines</parameter>\n\n</function>\n\n\n</tool_call>\n\n\n<IMPORTANT>\n\n- Function calls MUST follow the specified  
format: an inner <function=...></function> block must be nested within <tool_call></tool_call> XML tags\n- DO  
NOT use function calls inside <think></think> tags.\n- The value enclosed between parameter tags is preserved  
exactly as-is, including newlines and spaces.\n\n\n</IMPORTANT>\n' }}\n {%- endif %}\n {{- '<|im_end|>' }}\n {%- for  
message in loop_messages %}\n {%- if message.content is string %}\n {%- set content = message.content %}\n {%-  
else %}\n {%- set content = render_content(message.content) %}\n {%- endif %}\n {%- if message.role ==  
'assistant'\n %}\n {%- if message.reasoning_content is string %}\n {%- set reasoning_content =  
message.reasoning_content %}\n {%- else %}\n {%- set reasoning_content = '' %}\n {%- if '</think>' in content  
%}\n {%- set reasoning_content = content.split('</think>')[0].split('<think>')[1] %}\n {%- set content =  
content.split('</think>')[1] %}\n {%- endif %}\n {%- endif %}\n {%- if (keep_all_reasoning or loop.index0 >  
ns.last_user_index) and reasoning_content -%}\n {{- '<|im_start|>' + message.role + '\n\n<think>' +  
reasoning_content + '</think>' + content }}\n {%- else %}\n {{- '<|im_start|>' + message.role + '\n\n<think></think>' +  
content }}\n {%- endif %}\n {%- if message.tool_calls is defined and message.tool_calls is iterable and
```

```

message.tool_calls | length > 0 %}\n {%- for tool_call in message.tool_calls %}\n {%- if tool_call.function is defined
%}\n {%- set tool_call = tool_call.function %}\n {%- endif %}\n {{- '<tool_call>\n<function=' + tool_call.name + '>\n'
}}\n {%- if tool_call.arguments is defined %}\n {%- for args_name, args_value in tool_call.arguments.items %}\n {{-
'<parameter=' + args_name + '>' }}\n {%- set args_value = args_value | tojson | safe if args_value is mapping or
(args_value is sequence and args_value is not string) else args_value | string %}\n {{- args_value }}\n {{- '</
parameter>\n' }}\n {%- endfor %}\n {%- endif %}\n {{- '</function>\n</tool_call>' }}\n {%- endfor %}\n {%- endif
%}\n {{- '<|im_end|>' }}\n {%- elif message.role == \"user\" %}\n {{- '<|im_start|>' + message.role + '\n' +
render_content(message.content) + '<|im_end|>' }}\n {%- elif message.role == \"system\" %}\n {{- '<|im_start|>' +
message.role + '\n' + render_content(message.content) + '<|im_end|>' }}\n {%- elif message.role == \"tool\" %}\n
{%- if loop.previtem and loop.previtem.role != \"tool\" %}\n {{- '<|im_start|>tool\n' }}\n {%- endif %}\n {{-
'<tool_response>\n' }}\n {{- render_content(message.content) }}\n {{- '\n</tool_response>\n' }}\n {%- if not
loop.last and loop.nextitem.role != \"tool\" %}\n {{- '<|im_end|>' }}\n {%- elif loop.last %}\n {{- '<|im_end|>' }}\n {%-
endif %}\n {%- else %}\n {{- '<|im_start|>' + message.role + '\n' + render_content(message.content) + '<|im_end|>'
}}\n {%- endif %}\n {%- endfor %}\n {%- if add_generation_prompt %}\n {{- '<|im_start|>assistant\n' }}\n {%- if not
enable_thinking -%}\n {{- '<think></think>' -}}\n {%- else -%}\n {{- \" \" -}}\n {%- endif -%}\n {%- endif
%}\n\n\"eos_token\": \"<|im_end|>\", \"pad_token\": \"</endofextl
>\", \"unk_token\": null}}, \"createdAt\": \"2026-04-27T13:37:38.000Z\", \"discussionsDisabled\": false, \"discussionsSorting\": \"recent
created\", \"downloads\": 170204, \"downloadsAllTime\": 1583227, \"id\": \"XiaomiMiMo/MiMo-
V2.5\", \"isLikedByUser\": false, \"availableInferenceProviders\":
[{\"provider\": \"novita\", \"modelStatus\": \"live\", \"providerStatus\": \"live\", \"providerId\": \"xiaomimimo/mimo-
v2.5\", \"task\": \"conversational\", \"adapterWeightsPath\": \"audio_tokenizer/model.safetensors\", \"features\":
{\"structuredOutput\": false, \"toolCalling\": true}, \"isCheapestPricingOutput\": false, \"isFastestThroughput\": true, \"isModelAuth
{\"provider\": \"featherless-ai\", \"modelStatus\": \"live\", \"providerStatus\": \"live\", \"providerId\": \"XiaomiMiMo/MiMo-
V2.5\", \"task\": \"conversational\", \"adapterWeightsPath\": \"audio_tokenizer/
model.safetensors\", \"isCheapestPricingOutput\": false, \"isFastestThroughput\": false, \"isModelAuthor\": false},
{\"provider\": \"deepinfra\", \"modelStatus\": \"live\", \"providerStatus\": \"live\", \"providerId\": \"XiaomiMiMo/MiMo-
V2.5\", \"task\": \"conversational\", \"adapterWeightsPath\": \"audio_tokenizer/model.safetensors\", \"features\":
{\"structuredOutput\": true, \"toolCalling\": true}, \"isCheapestPricingOutput\": true, \"isFastestThroughput\": false, \"isModelAuthc
generation\", \"library_name\": \"transformers\", \"librariesOther\": [], \"trackDownloads\": true, \"model-
index\": null, \"evalResults\": [{\"dataset\": {\"id\": \"ScaleAI/SWE-
bench_Pro\", \"isBenchmark\": true, \"task_id\": \"SWE_Bench_Pro\", \"value\": 56.1, \"source\": {\"url\": \"https://huggingface.co/
XiaomiMiMo/MiMo-V2.5\", \"name\": \"Model Card\", \"isExternal\": false}, \"filename\": \".eval_results/swe-
bench_pro.yaml\", \"verified\": false, \"pullRequest\": 1, \"rank\": 24, \"label\": \"SWE Bench Pro\"}, {\"dataset\":
{\"id\": \"harborframework/terminal-bench-2.0\", \"isBenchmark\": true, \"task_id\": \"terminalbench_2\", \"value\": 65.8, \"source\":
{\"url\": \"https://huggingface.co/XiaomiMiMo/MiMo-V2.5\", \"name\": \"Model
Card\", \"isExternal\": false}, \"filename\": \".eval_results/terminal-
bench-2.0.yaml\", \"verified\": false, \"pullRequest\": 1, \"rank\": 8, \"sizeRank\":
{\"rank\": 1, \"value\": 500000000000, \"id\": \"500B\", \"label\": \"< 500B\"}, \"label\": \"Terminalbench 2\"}, {\"dataset\":
{\"id\": \"InternScience/ResearchClawBench\", \"isBenchmark\": false, \"task_id\": \"overall\", \"value\": 16.91, \"source\":
{\"url\": \"https://huggingface.co/XiaomiMiMo/MiMo-V2.5\", \"name\": \"Model
Card\", \"isExternal\": false}, \"filename\": \".eval_results/
researchclawbench.yaml\", \"verified\": false, \"pullRequest\": 17, \"rank\": 5, \"sizeRank\":
{\"rank\": 1, \"value\": 500000000000, \"id\": \"500B\", \"label\": \"< 500B\"}, \"label\": \"Overall\", \"notes\": \"ResearchHarness
evaluation with tools enabled, code execution, and a file-system workspace; completed 39/40
ResearchClawBench tasks.\"}, {\"dataset\": {\"id\": \"claw-eval/Claw-
Eval\", \"isBenchmark\": true, \"task_id\": \"general\", \"value\": 62.1, \"source\": {\"url\": \"https://claw-
eval.github.io\", \"name\": \"Claw-Eval Leaderboard\", \"isExternal\": true, \"author\":
{\"_id\": \"6038d6d0612f5eef3cc05ea9\", \"avatarUrl\": \"https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg\", \"fullname\": \"Lei
Li\", \"name\": \"tobiaslee\", \"type\": \"user\", \"isPro\": false, \"isHf\": false, \"isHfAdmin\": false, \"isMod\": false, \"followerCount\": 30, \"isUse
claw_eval.yaml\", \"verified\": false, \"pullRequest\": 10, \"rank\": 14, \"label\": \"General\", \"notes\": \"Pass% | N=3 | 161 tasks\"},
{\"dataset\": {\"id\": \"claw-eval/Claw-Eval\", \"isBenchmark\": true, \"task_id\": \"multimodal\", \"value\": 23.8, \"source\":
{\"url\": \"https://claw-eval.github.io\", \"name\": \"Claw-Eval Leaderboard\", \"isExternal\": true, \"author\":
{\"_id\": \"6038d6d0612f5eef3cc05ea9\", \"avatarUrl\": \"https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrlIZQ1f.jpeg\", \"fullname\": \"Lei
Li\", \"name\": \"tobiaslee\", \"type\": \"user\", \"isPro\": false, \"isHf\": false, \"isHfAdmin\": false, \"isMod\": false, \"followerCount\": 30, \"isUse

```

```

claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 3, "sizeRank":
{"rank": 3, "value": 500000000000, "id": "500B", "label": "< 500B", "label": "Multimodal", "notes": "Pass³% | N=3 | 101
tasks"}, {"dataset": {"id": "claw-eval/Claw-Eval", "isBenchmark": true, "task_id": "multi_turn"}, "value": 63.2, "source":
{"url": "https://claw-eval.github.io", "name": "Claw-Eval Leaderboard", "isExternal": true, "author":
{"_id": "6038d6d0612f5eef3cc05ea9", "avatarUrl": "https://cdn-avatars.huggingface.co/v1/production/
uploads/6038d6d0612f5eef3cc05ea9/ryhvAX5djQpD5OrIIZQ1f.jpeg", "fullName": "Lei
Li", "name": "tobiaslee", "type": "user", "isPro": false, "isHf": false, "isHfAdmin": false, "isMod": false, "followerCount": 30, "isUse
claw_eval.yaml", "verified": false, "pullRequest": 10, "rank": 3, "sizeRank":
{"rank": 1, "value": 500000000000, "id": "500B", "label": "< 500B", "label": "Multi Turn", "notes": "Pass³% | N=3 | 38
tasks"}}, {"private": false, "repoType": "model", "gated": false, "tags": ["transformers", "safetensors", "mimo_v2", "text-
generation", "multimodal", "vision-language", "audio", "agent", "video-understanding", "long-
context", "conversational", "custom_code", "en", "zh", "license:mit", "eval-results", "fp8", "region:us"], "tag_objs":
[{"id": "text-generation", "label": "Text Generation", "type": "pipeline_tag", "subType": "nlp"},
{"id": "transformers", "label": "Transformers", "type": "library"}, {"id": "safetensors", "label": "Safetensors", "type": "library"},
{"id": "en", "label": "English", "type": "language"}, {"id": "zh", "label": "Chinese", "type": "language"},
{"id": "mimo_v2", "label": "mimo_v2", "type": "other", "clickable": true},
{"id": "multimodal", "label": "multimodal", "type": "other", "clickable": true}, {"id": "vision-language", "label": "vision-
language", "type": "other", "clickable": true}, {"id": "audio", "label": "audio", "type": "other", "clickable": true},
{"id": "agent", "label": "agent", "type": "other", "clickable": true}, {"id": "video-understanding", "label": "video-
understanding", "type": "other", "clickable": true}, {"id": "long-context", "label": "long-
context", "type": "other", "clickable": true},
{"id": "conversational", "label": "conversational", "type": "other", "clickable": true},
{"id": "custom_code", "label": "custom_code", "type": "other", "clickable": true}, {"id": "eval-results", "label": "Eval
Results", "type": "other", "clickable": true}, {"id": "fp8", "label": "fp8", "type": "other", "clickable": true},
{"id": "license:mit", "label": "mit", "type": "license"}, {"type": "region", "label": "🌐 Region:
US", "id": "region:us"}], "transformersInfo":
{"auto_model": "AutoModelForCausalLM", "custom_class": "modeling_mimo_v2.MiMoV2ForCausalLM", "pipeline_tag": "
generation"}, {"widgetData": [{"text": "Hi, what can you help me with?"}, {"text": "What is 84 * 3 / 2?"}, {"text": "Tell me an
interesting fact about the universe!"}, {"text": "Explain quantum computing in simple terms."}], "safetensors":
{"parameters":
{"F32": 68012416, "BF16": 4052024960, "F8_E4M3": 306655002624}, {"total": 310775040000, "sharded": true, "totalFileSize
Evaluation results ScaleAI/SWE-bench_Pro · SWE Bench Pro View evaluation results leaderboard 56.1

```

harborframework/terminal-bench-2.0 · Terminalbench 2 View evaluation results leaderboard 65.8

InternScience/ResearchClawBench · Overall View evaluation results 16.91 *

claw-eval/Claw-Eval leaderboard

General View evaluation results source Pass³% | N=3 | 161 tasks 62.1 *

Multimodal View evaluation results source Pass³% | N=3 | 101 tasks 23.8 *

Multi Turn View evaluation results source Pass³% | N=3 | 38 tasks 63.2 *