LLM Infrastructure Cheat Sheet

This document outlines all 20 models currently available on your Ollama server (ollama.ldmathes.cc/v1/models), categorized by access method (Cloud or Local) and their primary strength.

The models marked **CLOUD** are large-scale, high-performance, and typically run on the remote GPU, offering faster responses and better reasoning for complex tasks. The **LOCAL** models run directly on your host machine, making them excellent for quick, low-latency, or vision-based tasks.

🚀Group 1: High-Performance Cloud Models (The Heavy Hitters)

These models are ideal for complex logical problems, large code repositories, or when you need state-of-the-art performance.

Model ID	Total Parameters (Approx.)	Key Feature / Architecture	Primary Use Case (When to Use It)
deepseek-v3.1:671b-cloud	671 Billion (MoE)	Max Reasoning & Complex Logic	**The Powerhouse:** Best for difficult, multi-step reasoning, advanced mathematics, and precision-critical tasks.
gpt-oss:120b-cloud	117 Billion (MoE)	Versatile Agent & Tool Use	**Best All-Rounder:** Excellent general reasoning, instruction following, and agentic capabilities rivaling proprietary models.
qwen3-coder:480b-cloud	480 Billion (MoE)	Elite Agentic Coding & 1M Context	**Coding Specialist:** Use for massive context/repository analysis, complex code generation, and sophisticated tool use.
kimi-k2-thinking:cloud	High (MoE)	Structured Planning & Thinking	**Meticulous Execution:** Use for tasks requiring deliberate, step-by-step reasoning and long-form, structured content generation.
glm-4.6:cloud	High	Fast General Agent	**High-Speed Versatility:** Ideal for a balance of speed and high-quality general response when a specific focus isn't needed.
minimax-m2:cloud	High	Efficient Code Agent	**Coding Efficiency:** Use for high-performance code generation and quick, clean scripting solutions.

💻Group 2: Local & Specialty Models (On-Device Efficiency)

These models run directly on your local Ollama host, making them fast for immediate interaction or specialized tasks like vision.

Vision Models

Model ID	Category	Parameters (Approx.)	Primary Use Case (When to Use It)
llava:13b	VISION (Multimodal)	13 Billion	**Image Analysis:** Use for interpreting, describing, or answering questions based on images you provide.

Code-Focused Local Models

Model ID	Category	Parameters (Approx.)	Primary Use Case (When to Use It)
deepseek-r1:latest	Code	~7 Billion	**Local Code Analysis:** Good for quick analysis of code snippets and suggesting localized changes or fixes.
qwen3-coder:latest	Code	~7B / 30B	**Efficient Local Coding:** The smaller, faster local variant for day-to-day coding tasks and function calling.
qwen2.5-coder:1.5b	Code	1.5 Billion	**Lightweight Coding:** Ultra-efficient coding assistant for quick code suggestions and simple programming tasks.

Function/Agent Models

Model ID	Category	Parameters (Approx.)	Primary Use Case (When to Use It)
firefunction-v2:latest	Function/ Agent	-	**Tooling & Function Calls:** Specialized in robust and reliable execution of function and tool-use prompts.

Chat & Instruction Models

Model ID	Category	Parameters (Approx.)	Primary Use Case (When to Use It)
notus:latest	Chat/ Instruction	~8 Billion	**Instruction Following:** Excellent for conversational prompts, maintaining persona, and following specific instructions closely.
alfred:latest	Chat	-	**General Conversational Chat:** A reliable option for creative writing, brainstorming, and casual dialogue.

General Purpose Local Models

Model ID	Category	Parameters (Approx.)	Primary Use Case (When to Use It)
mistral:latest	General	7 Billion	**Fast Local Response:** The go-to choice for speed; use for basic, low-latency text completion and summarization.
phi3:mini	General	~3.8 Billion	**Compact Intelligence:** Powerful small model with strong reasoning for its size; ideal for resource-constrained environments.

Ultra-Efficient Models

Model ID	Category	Parameters (Approx.)	Primary Use Case (When to Use It)
gemma3:4b	Efficient	4 Billion	**Efficiency/Small Tasks:** Use on devices with limited resources, or for quick, simple tasks where high accuracy is not paramount.
gemma2:2b	Efficient	2 Billion	**Lightweight Tasks:** Fast, efficient model for straightforward queries and basic text processing.
qwen2.5:1.5b	Efficient	1.5 Billion	**Quick Responses:** Extremely fast for simple completions, translations, and basic Q&A.
gemma3:1b	Efficient	1 Billion	**Ultra-Fast/Minimal:** Fastest local model for simple checks, basic translation, and minimal memory usage.
tinyllama:latest	Efficient	~1.1 Billion	**Ultra-Minimal:** The smallest option for extremely resource-limited scenarios and basic text operations.

📊 Quick Selection Guide

Need maximum reasoning power? → Use **deepseek-v3.1:671b-cloud**

Need best all-around performance? → Use **gpt-oss:120b-cloud**

Working with large codebases? → Use **qwen3-coder:480b-cloud**

Need structured, step-by-step thinking? → Use **kimi-k2-thinking:cloud**

Need to analyze images? → Use **llava:13b**

Quick local coding help? → Use **qwen3-coder:latest** or **qwen2.5-coder:1.5b**

Need speed above all else? → Use **tinyllama:latest**, **gemma3:1b**, or **qwen2.5:1.5b**

Balanced local performance? → Use **mistral:latest** or **phi3:mini**