Open-Source LLM Selection Cheatsheet

For local, private use: choosing and evaluating models for RAG & agents on your own machine.

Think in tradeoffs: *capability* ↔ *speed* ↔ *memory* ↔ *licensing* ↔ *specialization*.

PRINT-FRIENDLY • 2 PAGES

LOCAL ONLY　PRIVACY-FIRST　MODEL SELECTION

🧠 1. Mental Model — How to Think About Any Model

Every LLM you'll touch is some balance of:

- **Capability** – reasoning, coherence, following instructions
- **Speed** – latency on *your* hardware
- **Memory footprint** – RAM / VRAM requirement
- **Licensing** – personal vs. commercial use
- **Specialization** – general chat, coding, embeddings, tiny models, etc.

> **Guiding question:** "Is this model good enough for my task while staying fast and light on this machine?"

Don't chase biggest　Start small, iterate

Scale only when needed

📏 2. Model Size vs. Behavior (Rough Intuition)

SIZE	TYPICAL BEHAVIOR / USE
3B	Very fast, basic reasoning. Good for tiny tasks, toys, or constrained use.
7B	Sweet spot for local chat/RAG. Feels like an earlier GPT-3.5 for many tasks.
13B	Better coherence & reasoning; slower but still usable for interactive work.
30B+	Stronger reasoning, "closer to GPT-4" feel. Heavy; only if hardware + latency allow.

On a modern Apple Silicon machine:
→ **Start at 7B**. Move to 13B when you hit capability limits. Only consider 30B+ if you accept slower responses.

🎯 3. Starter Models by Use Case

GENERAL CHAT / RAG DEFAULTS

Everyday “brain” for private assistants & RAG:

- Llama 3 8B Instruct
- Mistral 7B Instruct

Use for: local chat, document Q&A, prototyping agents.

MORE REASONING POWER

When 7B isn’t quite enough:

- Mixtral 8x7B
- Larger Llama/Qwen variants

Use if you need stronger reasoning and can tolerate more latency.

CODING-HEAVY WORK

When code quality is central:

- DeepSeek Coder 6.7B
- Codestral / Qwen Coder

TINY / ULTRA-LIGHTWEIGHT

For constrained hardware & quick tests:

- Phi-3 Mini / Phi-2
- TinyLlama

Great for experiments, mobile, or embedded use cases.

EMBEDDINGS FOR RAG

Text embedding workhorses:

- nomic-embed-text
- bge-m3
- all-MiniLM

⚙️ 4. Quantization Cheatsheet (Local Formats)

Quantization compresses models → less memory, more speed, small quality tradeoff.

FORMAT	WHEN TO USE
Q4_K_M	Smallest, fastest. Great for MVPs and trying many models quickly.
Q5_K_M	Balanced. Good default for everyday development and testing.
Q8	Heavier but higher-fidelity. Use when accuracy matters more than speed.

> **Rule of thumb:** Start with **Q4/Q5** for exploration. Move to **Q8** only if you see quality issues and your hardware can handle it.

📜 5. Licensing Sanity Check

For private local experiments: almost everything is fine. For commercial apps:

- Confirm commercial use is allowed
- Check if attribution is required
- Note any restrictions for large companies

> Keep a small note in your repo: *“This prototype uses Model X; confirm license before production.”*

Model Evaluation & Comparison Sheet

WORKING DOC • FILL BY HAND

Use this page while you experiment with different open-source models on your local stack.
Same hardware, same tasks, same prompts → comparable results.

🧩 1. Context for This Evaluation

Fill this in once per evaluation session.

Hardware	______________________________
Runtime	______________________________
Project / Use Case	______________________________
Docs / Data	______________________________

Example: “Mac mini M4, 24 GB • Ollama • Private RAG over EDM guidance docs.”

📚 2. Models Under Test (Inventory)

List each model + key settings so you can reproduce results later.

MODEL NAME	SIZE	QUANTIZATION	CONTEXT WINDOW	NOTES (E.G., INSTRUCT / CODER)
______________	_____	_____	_____	______________________________
______________	_____	_____	_____	______________________________
______________	_____	_____	_____	______________________________

🧪 3. Tasks / Prompts Used for Comparison

Use the same prompts for every model. You can sketch them here or reference a notebook.

- General explanation: ______________________________
- Domain-specific reasoning: __________________________
- RAG grounding (same context for each): ___________________
- Coding / tool-use task: ____________________________
- Edge case / hallucination check: ______________________

You can also glue a printed prompt sheet or QR link here if you prefer.

📊 4. Comparison Matrix (1–5 or Notes)

For each model, score 1–5 or jot quick notes for each dimension. Use the same tasks above.

MODEL	LATENCY	REASONING	GROUNDING	HALLUCINATIONS	STYLE / FIT	NOTES
____________	_____	_____	_____	_____	_____	____________________
____________	_____	_____	_____	_____	_____	____________________
____________	_____	_____	_____	_____	_____	____________________

Dimensions: **Latency** (speed), **Reasoning** (quality of answers), **Grounding** (uses provided context correctly), **Hallucinations** (confident wrongness), **Style / Fit** (does it feel right for this app?).

🏁 5. Final Choice for This Project

Default "brain" for this app:

Model: ________________________________

Why it wins (2–3 bullets):