Mit Vibe-Coding und LLMs gegen Bürokratie

Paperless-ngx und lokale LLMs

Ben • 2025/10/16 • Heroku/Salesforce

Warum machen ~~wir~~ ich das?

Halbwegs gut gelaunt durch den Juli kommen (Steuererklärung)

Workflow für

- Brief/E-Mail landet im Postfach/Briefkasten und wie finde ich ihn wieder

Zentralisiertes Organisieren und Indexieren von Dokumenten, um sie schneller aufzufinden

Paperless-ngx

docs.paperless-ngx.com

open-source Dokumentenmanagement (DMS)

OCR Unterstützung (via Tesseract)

Maschinelles Lernen um Tags, Absender und Typ zu erkennen*

Integrierte Workflows um Paperless-ngx zu automatisieren

E-Mail Unterstützung (Weiterleiten an secret@abwesend.com)

Ok-ish iOS app (github.com/paulgessinger/swift-paperless)

Closed-Weight LLMs

GPT-(2,3,4) von OpenAI

Claude Sonnet/Opus von Anthropic

Gemini von Google

Grok-3 von xAI (open-weight angekündigt)

Nutzbar via

- Provider API (OpenAI API, Anthropic API etc.)
- OpenRouter.ai
- AWS Bedrock, Azure AI oder VertexAI (Google), etc

Lokale LLMs

ChatGPT ist viel einfacher zu nutzen

Warum alles lokal?

- Meine Dokumente sollen nicht bei OpenAI landen.
- Lokalle LLMs sind gut genug
- Fürs Gewissen? Nur 240W statt 14.7 kWh

Lokale LLMs

Open-Weight Modelle

Huggingface ist GitHub für LLM’s

LLaMA (1,2,3) von Meta, **DeepSeek** (V3, R1) von DeepSeek AI, **Grok-(1,2)** von xAI

Gemma von Google DeepMind, **Qwen** (2.5, 3) von Alibaba Cloud

Mistral / Mixtral von Mistral AI, **Kimi K2** von Moonshot AI, **GLM-4.5** von Z.ai

Dense Modell (LLaMA, Mistral, Falcon, Gemma, Qwen, GPT-3/4)

Alle Parameter werden aktiviert, dadurch wird viel mehr Arbeitsspeicher gebraucht

MoE (Mixture of Experts) Modell (Mixtral, DeepSeekMoE, Qwen-MoE)

Kann Parameter selektiv laden/aktivieren, braucht weniger Arbeitsspeicher

Lokale LLMs

Software

Was machen wir hier?

- Kein Training
- Inferenz (“anwenden" des Modells, Eingaben verarbeiten und Ausgaben generieren)

Wozu?

- Kann Modell Dateiformat lesen (GGUF am populärsten)
- Stellt HTTP Endpunkt bereit (OpenAI API spec)
  - Beispiel Chat API /v1/responses, /v1/chat/completions, /v1/embeddings, /v1/completions

Gut zum Starten sind Ollama und LMStudio (closed source)

Besser vLLM und LLaMA.cpp (open source)

Gute Quelle: Subreddit https://old.reddit.com/r/LocalLLaMA

Hardware (Inferenz)

Was brauch ich? Speziell für Inference

VRAM (pro Milliarde Parameter) + 20% für Overhead (Kontext etc)

- FP16 ~2 GB
- INT8 ~1 GB
- 4-Bit (Q4_K_M, GPTQ etc) ~0,5 GB

Modifizierte NVIDIA RTX 5090 mit bis zu 128 GB VRAM auf AliBaba/AliExpress

- RTX 4090 modifiziert von 24 GB auf 48 GB
- GN https://youtu.be/1H3xQaf7BFI?t=9292
- GN https://youtu.be/1H3xQaf7BFI?t=9806

Hardware (Inferenz)

Was brauch ich? Speziell für Inference

Speicherbandbreite

NVIDIA Blackwell B200 (HBM3e) bis zu 8 TB/s

NVIDA Hopper H100 (HBM3) 3,5 TB/s

NVIDIA RTX 5090 mit 28 GB GDDR7 bis zu 2 TB/s

Apple M4 Max 128 GB bis zu 546 GB/s

Apple M4 32 GB bis zu 120 GB/s

Hardware

Apple

Apple M5 wurde gestern angekündigt

- “over 4x peak GPU performance over M4” - deutet auf “matmul" in Apple GPU hin
- 30% mehr Speicherbandbreiten (~153 GB/s vs ~120 GB/s)
- Eventuell auch +30% für M5 Pro/Max?
- https://www.apple.com/newsroom/2025/10/apple-unleashes-m5-the-next-big-leap-in-ai-performance-for-apple-silicon/

Hardware

Optionen für zuhause

Dual NVIDIA RTX 3090, 4090, 5090 + Ältere AMD Threadripper Generation

Apple Mac mini / Studio

- Eher auf nächste Generation warten (Mac mini M5 oder Mac Studio M5 Ultra)
- Mac mini mit 32 GB Arbeitsspeicher (Reuegefühle)

Hardware

NVIDIA

NVIDIA DGX Spark

- 128 GB gemeinsamer Arbeitsspeicher Speicherbandbreite 273 GB/s
- NVIDIA ConnectX-7 (2x100 Gbit/s Interconnect)
- Eher für Research und nicht für lokale Inferenz (viel VRAM, wenig Bandbreite, Ecosystem)
- ~4000 Euro
- Level1Techs https://www.youtube.com/watch?v=Lqd2EuJwOuw

Hardware

Rechenzentrum

Typisch

- 8 x NVIDIA H100 pro Server (ca ~$30.000 je Karte)
- 10 x Server pro Rack
- Ungefähr $3.000.000 pro Rack

Neu

- NVIDIA DGX B200 ($500.000)
- https://www.nvidia.com/en-us/data-center/dgx-b200/

Hardware

NVIDIA, NVIDIA, NVIDIA, NVIDIA, AMD?

Hardware ist vergleichbar

Software und Ökosystem verhindern den Erfolg

- NVIDA CUDA Framework ist überall und “works out of the box”
  - PyTorch etc sind optimiert für CUDA
  - Docker Images für alles verfügbar
- AMD ROCm kommt sehr langsam in Fahrt
  - AMD musste erst Intel beseitigen
  - Eventuell jetzt AMD Aktien kaufen?! (Keine Finanzberatung :-)

Paperless-ngx mit LLMs erweitern

github.com/clusterzx/paperless-ai

Dokumentanalyze mit Hilfe der Paperless-ngx API

- Eingebetteten Inhalt des PDFs
- Systemprompt

Disclaimer: Paperless-AI scheint aufgegeben worden sein

- <think> Tags werden nicht korrekt behandelt

Alternative: github.com/icereed/paperless-gpt

Scan Drop

Vibe-coded Fake FTP

Mein Workflow

- Papier in Scanner legen
- Scanner lädt Dokument via FTP zu Scan Drop
- Scan Drop lädt Dokument zu Paperless-ngx via HTTP API

Alternative

- Scanner legt Dokument in einen speziellen Ordner (mit FTP oder Samba)
- Paperless-ngx überwacht den Ordner

Links

https://docs.paperless-ngx.com
https://github.com/ggml-org/llama.cpp
https://github.com/vllm-project/vllm
https://old.reddit.com/r/LocalLLaMA
https://lmstudio.ai
https://huggingface.co
https://github.com/clusterzx/paperless-ai
https://github.com/icereed/paperless-gpt
https://github.com/paulgessinger/swift-paperless
https://github.com/beanieboi/scan-drop