Lokale LLM - ChatGPT ohne Cloud

Chatbots wie ChatGPT, Microsoft Copilot oder Google Gemini sind aus dem Alltag nicht mehr wegzudenken. Sie unterstützen uns bei der Texterstellung, beantworten Fragen und helfen, komplexe Aufgaben zu bewältigen. Während viele Nutzer auf cloudbasierte Lösungen zurückgreifen, also Software, die über das Internet auf externe Rechenzentren zugreift, gibt es zunehmend Alternativen, die direkt auf dem eigenen Computer betrieben werden können. Diese lokalen Lösungen bieten eine individuellere Nutzung, ermöglichen mehr Kontrolle über die eigenen Daten und tragen maßgeblich zur Einhaltung hoher Datenschutzstandards bei. Dank fortschreitender Technologie und leistungsfähigerer Hardware können heute auch ressourcenintensive KI-Modelle auf Consumer-Geräten effizient genutzt werden.

Lokale LLMs: Funktionsweise, Vorteile und Herausforderungen

Große Sprachmodelle (Large Language Models, LLMs) sind KI-gestützte Systeme, die auf enormen Mengen an Textdaten trainiert wurden. Sie analysieren, interpretieren und generieren natürliche Sprache mit beeindruckender Präzision. Im Gegensatz zu cloudbasierten Lösungen, die eine ständige Internetverbindung erfordern, laufen lokale LLMs direkt auf dem eigenen Computer. Dies bietet eine vollständig unabhängige Nutzung und erhöht die Sicherheit. Zentrale Technologien für den Betrieb lokaler LLMs sind die Inferenz – die eigentliche Textverarbeitung – und die Quantisierung zur Reduzierung der Modellgröße für ressourcenschonenden Betrieb. Fortschritte in der Modelloptimierung ermöglichen es heute, leistungsfähige Modelle auch auf Standard-Hardware zu betreiben, wodurch sich der Zugang zu dieser Technologie erheblich vereinfacht.

Ein entscheidender Vorteil des lokalen Betriebs von LLMs ist der Datenschutz und die Datensicherheit. Alle Daten verbleiben auf dem eigenen System, wodurch das Risiko ungewollter Datenlecks minimiert wird. Dies ist besonders in Bereichen mit hohen Datenschutzanforderungen von Vorteil. Zudem sind lokale LLMs unabhängig von einer Internetverbindung nutzbar, was sie zu einer idealen Lösung für Arbeitsumgebungen mit eingeschränktem oder sensiblem Zugriff macht. Darüber hinaus können langfristig durch den Verzicht auf wiederkehrende Cloud-Abonnementgebühren erhebliche Kosten eingespart werden.

Trotz dieser Vorteile gibt es einige Herausforderungen. Die benötigte Hardwarekapazität, insbesondere in Bezug auf Arbeitsspeicher und Rechenleistung, stellt für viele Nutzer eine Hürde dar. Leistungsstarke Modelle erfordern spezialisierte Hardware wie GPUs, um eine flüssige Inferenzgeschwindigkeit zu gewährleisten. Zudem sind lokale LLMs möglicherweise nicht so leistungsfähig oder benutzerfreundlich wie einige

cloudbasierte Alternativen. Während Modelle, die auf einfachen Office-Laptops laufen, nicht dieselbe Wortgewandtheit oder analytischen Fähigkeiten wie GPT-4o mit seinen geschätzten 1,8 Billionen trainierbaren Parametern und einer Modellgröße von mehreren Terabyte an GPU-Speicher (Quelle: KLU AI) erreichen, bieten kleinere Open-Source-Modelle wie Llama3.x oder Microsofts Phi [Platzhalter für Vergleich mit anderen Open-Source-Modellen] dennoch beeindruckende Leistung. Sie sind eine interessante Alternative für viele Anwendungsfälle und können dazu beitragen, Aufgaben ohne Cloud-Unterstützung sicher und datenschutzfreundlich umzusetzen.

Bevor ein LLM lokal betrieben wird, sollten wesentliche Aspekte wie Hardware-Anforderungen, Datenschutz, Modellwahl, Software-Tools und Kosten sorgfältig geprüft werden. Die folgenden Fragen dienen als Orientierung, um eine fundierte Entscheidung zu treffen:

- Welche Hardware-Ressourcen stehen zur Verfügung, und reichen sie für den effizienten Betrieb eines LLMs aus?
- Welche Datenschutz- und Sicherheitsanforderungen gibt es, und ist ein lokales LLM dafür die beste Lösung?
- Welches Modell passt am besten zu den Anforderungen hinsichtlich Performance, Speicherbedarf und Anwendungsfall?
- Welche Software und Tools werden für Installation, Verwaltung und Nutzung des LLMs benötigt?
- Welche Kosten entstehen durch den lokalen Betrieb, und wie verhält sich das im Vergleich zu einer cloudbasierten Lösung?

Praktische Werkzeuge für den lokalen Einsatz von LLMs

Für den praktischen Einsatz lokaler LLMs stehen verschiedene Tools zur Verfügung, die eine einfache Installation und Nutzung ermöglichen:

- **Ollama** ist eine Plattform zur Verwaltung verschiedener LLMs auf lokalen Rechnern. Mittels Kommandozeilenschnittstelle und API-Unterstützung können Entwickler LLMs problemlos in eigene Projekte integrieren und flexibel zwischen verschiedenen Modellen wechseln.

- **GPT4All** ermöglicht die Nutzung leistungsstarker LLMs auf nahezu jedem Computer. Die Plattform bietet eine Auswahl an vortrainierten Modellen, die für unterschiedliche Hardwareanforderungen optimiert sind, und richtet sich sowohl an Einsteiger als auch an Fortgeschrittene.

- **Anything LLM** erlaubt es, individuelle KI-Assistenten mit lokalem Backend zu erstellen. Es bietet erweiterte Funktionen wie Dokumentenanalyse und Langzeit-Konversationsspeicherung, was es besonders für professionelle Anwendungen interessant macht.

- **Jan AI** bietet eine grafische Benutzeroberfläche und erleichtert den Einstieg in die Welt der lokalen KI-Lösungen. Diese Plattform unterstützt verschiedene Modelle und Anwendungsfälle, darunter Textverarbeitung und komplexere Aufgaben wie die Analyse von Programmcode.

Voraussetzungen für die Nutzung von Ollama

Hardware:

- PC mit Windows 10/11 (Windows 10 Version 1903 oder neuer) oder Windows 11
- Dedizierte Grafikkarte wird empfohlen für die lokale Ausführung anspruchsvollerer LLMs.

Software:

- Ein Webbrowser für den Zugriff auf die Ollama-Plattform.

Schritt-für-Schritt-Anleitung zur Nutzung von Ollama

Schritt 0: Wichtige Ollama Befehle

- `ollama list` Zeigt alle verfügbaren lokalen Modelle an.
- `ollama pull llama3.1` Lädt das Modell **llama3.1** herunter.
- `ollama run llama3.1` (Lädt ggf. und) Startet das Modell **llama3.1**.
- `ollama rm llama3.1` Entfernt das Modell **llama3.1** von dem System.
- `ollama ps` Zeigt alle derzeit gestarteten Modelle.

Schritt 1: Download und Installation

1. **Website besuchen:**
    - Öffnen Sie Ihren Webbrowser und gehen Sie auf die offizielle Website von Ollama unter ollama.com.
2. **Passende Version herunterladen:**
    - Wählen Sie je nach Betriebssystem (Windows, Mac oder Linux) die passende Version aus.
    - Klicken Sie auf den Download-Button und warten Sie, bis der Download abgeschlossen ist.

3. **Installation durchführen (Windows):**
    - Navigieren Sie zu Ihrem Download-Ordner und doppelklicken Sie auf die heruntergeladene Datei (z. B. ollama-installer.exe).
    - Folgen Sie den Anweisungen des Installationsassistenten und wählen Sie gegebenenfalls ein Installationsverzeichnis.
    - Falls erforderlich, installieren Sie die Visual Studio Runtime.
    - Nach Abschluss der Installation bestätigen Sie den Abschluss mit „Fertigstellen“.

Schritt 2: Auswahl eines Modells

Ollama bietet eine Vielzahl von Open-Source-Modellen zur Auswahl. Um ein Modell auszuwählen:

1. **Liste der verfügbaren Modelle anzeigen:**
    - Öffnen Sie das Terminal oder die Eingabeaufforderung (Windows: cmd.exe, Mac/Linux: Terminal).
    - Geben Sie den folgenden Befehl ein:

```
ollama list
```

    - Dies zeigt eine Liste aller auf dem PC verfügbaren Modelle an (anfangs leer).
2. **Empfohlene Modelle:**

    Auf https://ollama.com/search sind unterschiedliche Modelle aufgelistet, die lokal ausgeführt werden können.

    Populäre Modelle sind:

    - **Llama 3.1 8B:** Leistungsstark für anspruchsvollere Anwendungen.
    - **Phi-3-5 3B:** Gut geeignet für logisches Denken und Mehrsprachigkeit.
    - **Llama 3.3 2B:** Effizient für Anwendungen mit begrenzten Ressourcen.
    - **Phi 4 14B:** State-of-the-art Modell mit erhöhter Hardware-Anforderung aber Leistung vergleichbar mit deutlich größeren Modellen.

Schritt 3: Ausführen des ersten Modells

Nach der Installation können Sie Ihr erstes Modell starten,

1. **Eingabeaufforderung oder Terminal öffnen:**
    - Unter Windows drücken Sie Win + R, geben cmd ein und bestätigen mit Enter.

- Auf Mac/Linux öffnen Sie das Terminal über das Anwendungsmenü.

2. **Modell herunterladen und ausführen:**
   - Beispiel für das Modell Llama 3.1:
   - `ollama run llama3.1`
   - Das Modell wird nun heruntergeladen und gestartet. Der Fortschritt wird im Terminal angezeigt.
3. **Überprüfung des Betriebs:**
   - Wenn das Modell erfolgreich gestartet ist, sehen Sie eine Eingabeaufforderung, in der Sie mit dem Modell chatten können.

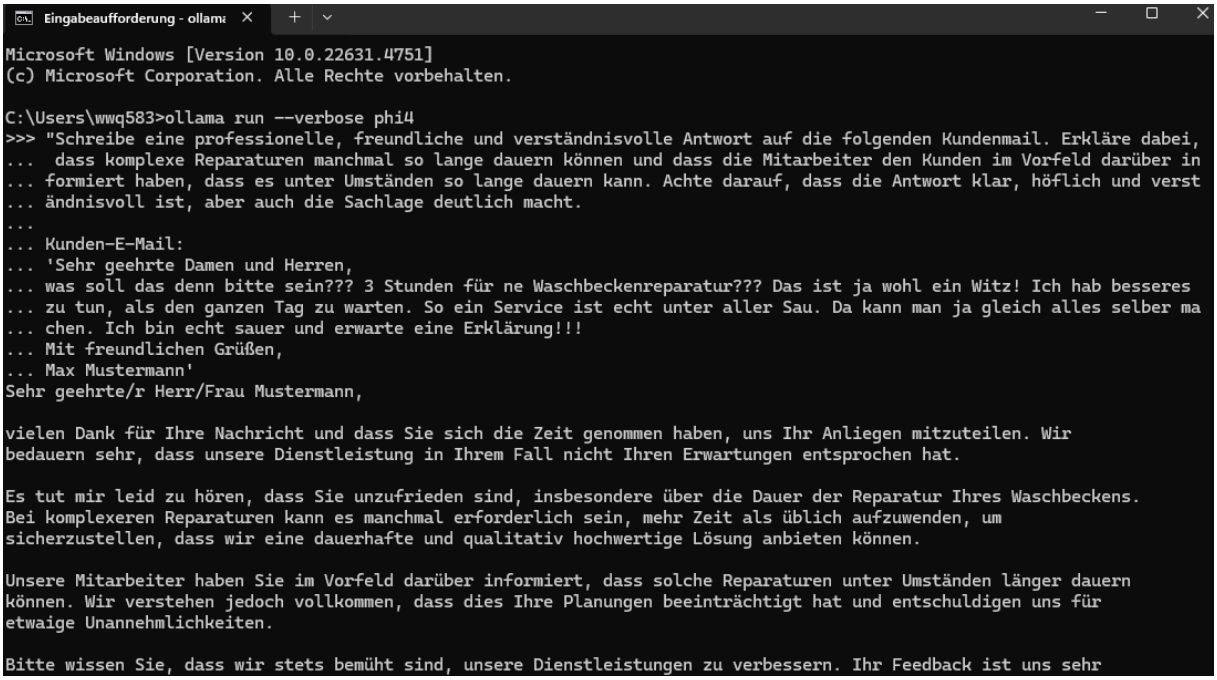

Abbildung 1 Ollama eingabe Aufforderung

Alternative GUI: Ollama UI Browsererweiterung

Zusätzlich zur Ausführung von LLMs im Terminal gibt es eine praktische Browsererweiterung namens **Ollama UI**, die eine benutzerfreundliche Oberfläche für die Interaktion mit den Modellen bietet. Diese Erweiterung ermöglicht es, Modelle auszuwählen, Konversationen im Browser zu führen und sie für die spätere Nutzung zu speichern.

- **Link zur Erweiterung:** Ollama UI für Chrome

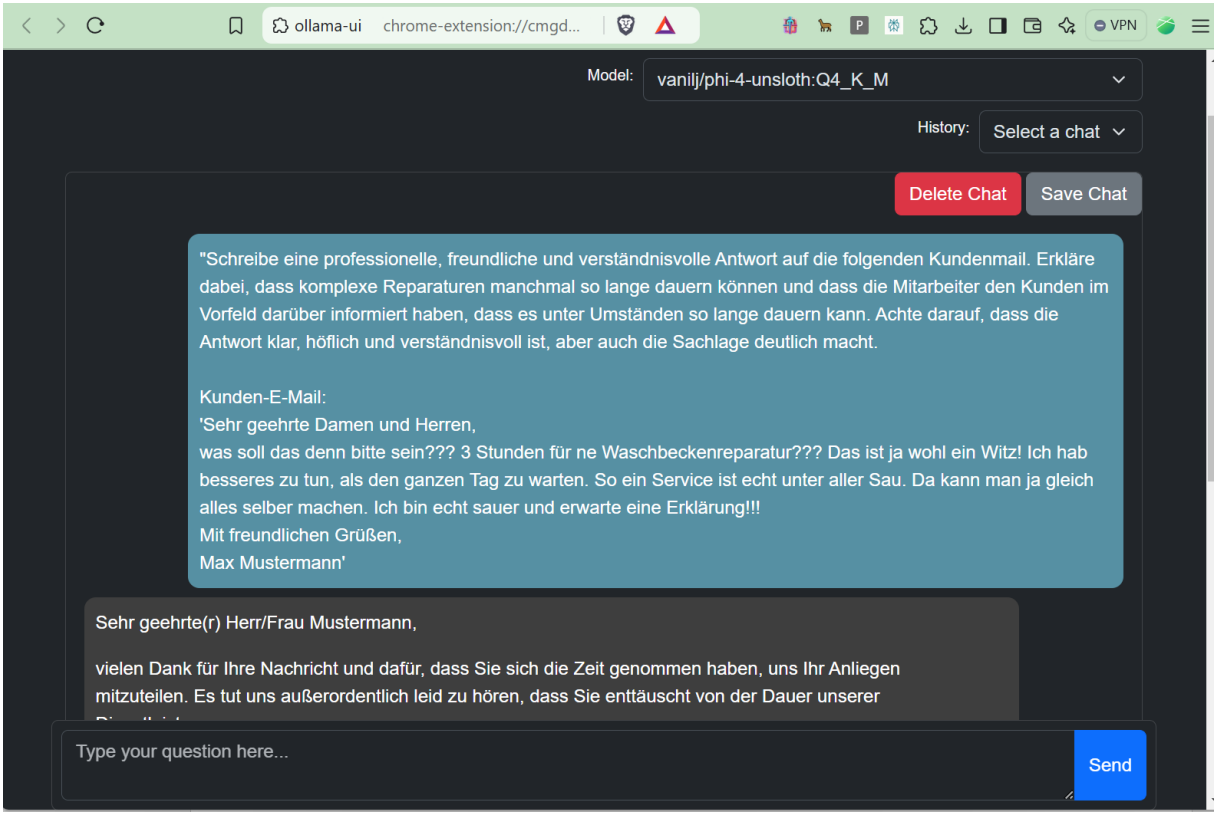

Abbildung 2 Ollama UI

GPT4All mit Llama 3 und RAG

GPT4All ist eine leistungsstarke Anwendung, die es ermöglicht, große Sprachmodelle wie Llama 3 lokal auf Ihrem Computer auszuführen und mit Ihren eigenen Dokumenten zu interagieren. In diesem Tutorial führen wir Sie durch den Prozess der Einrichtung und Nutzung von GPT4All mit Fokus auf die RAG-Funktionalität (Retrieval-Augmented Generation) über LocalDocs.

Was ist Retrieval-Augmented Generation (RAG)?

Retrieval-Augmented Generation (RAG) ist eine Technik der künstlichen Intelligenz, die Sprachmodelle mit externen Wissensquellen kombiniert, um präzisere und kontextbezogene Antworten zu liefern. Der Prozess beginnt mit der Verarbeitung von Rohdaten zu Vektorrepräsentationen, die in einer Vektordatenbank gespeichert werden. Dieser Schritt ist nur einmalig oder bei Datenaktualisierungen erforderlich.

Die Schritte des RAG-Prozesses

Schritt 0: Aus den vorhandenen Rohdaten werden Embeddings erzeugt und in einer Vektordatenbank abgelegt. Dieser Schritt ist nur notwendig, wenn neue Daten hinzugefügt oder bestehende Daten aktualisiert werden müssen.

Schritt 1: Der Prozess startet, sobald ein Benutzer eine Anfrage stellt. Diese Anfrage wird an das RAG-Framework übermittelt und bildet die Grundlage für die anschließende Verarbeitung.

Schritt 2: Die Anfrage wird in eine Vektorform umgewandelt und mit den gespeicherten Embeddings in der Vektordatenbank verglichen. Die semantische Suche ermittelt die relevantesten Informationen, die inhaltlich zur Anfrage passen.

Schritt 3: Die gefundenen relevanten Informationen werden der ursprünglichen Anfrage hinzugefügt. Dieser angereicherte Prompt wird verwendet, um eine präzisere und kontextuelle Antwort zu ermöglichen.

Schritt 4: Das Sprachmodell verarbeitet den erweiterten Prompt und generiert eine detaillierte und fundierte Antwort, die dem Benutzer zurückgegeben wird.

Der Vorteil von RAG liegt darin, dass es den Zugriff auf aktuelle und spezifische Informationen ermöglicht, ohne das Sprachmodell erneut trainieren zu müssen. Dies spart Ressourcen und verbessert die Qualität der generierten Inhalte erheblich. RAG wird in Bereichen wie Kundenservice, Forschung und unternehmensinternem Wissensmanagement eingesetzt, um relevante Informationen aus großen Datenmengen effizient nutzbar zu machen.

Installation und Einrichtung

GPT4All herunterladen und installieren

1. Besuchen Sie die offizielle GPT4All-Website.
2. Laden Sie die für Ihr Betriebssystem passende Version herunter (Windows, Mac oder Linux).
3. Führen Sie die Installation aus und folgen Sie den Anweisungen auf dem Bildschirm.

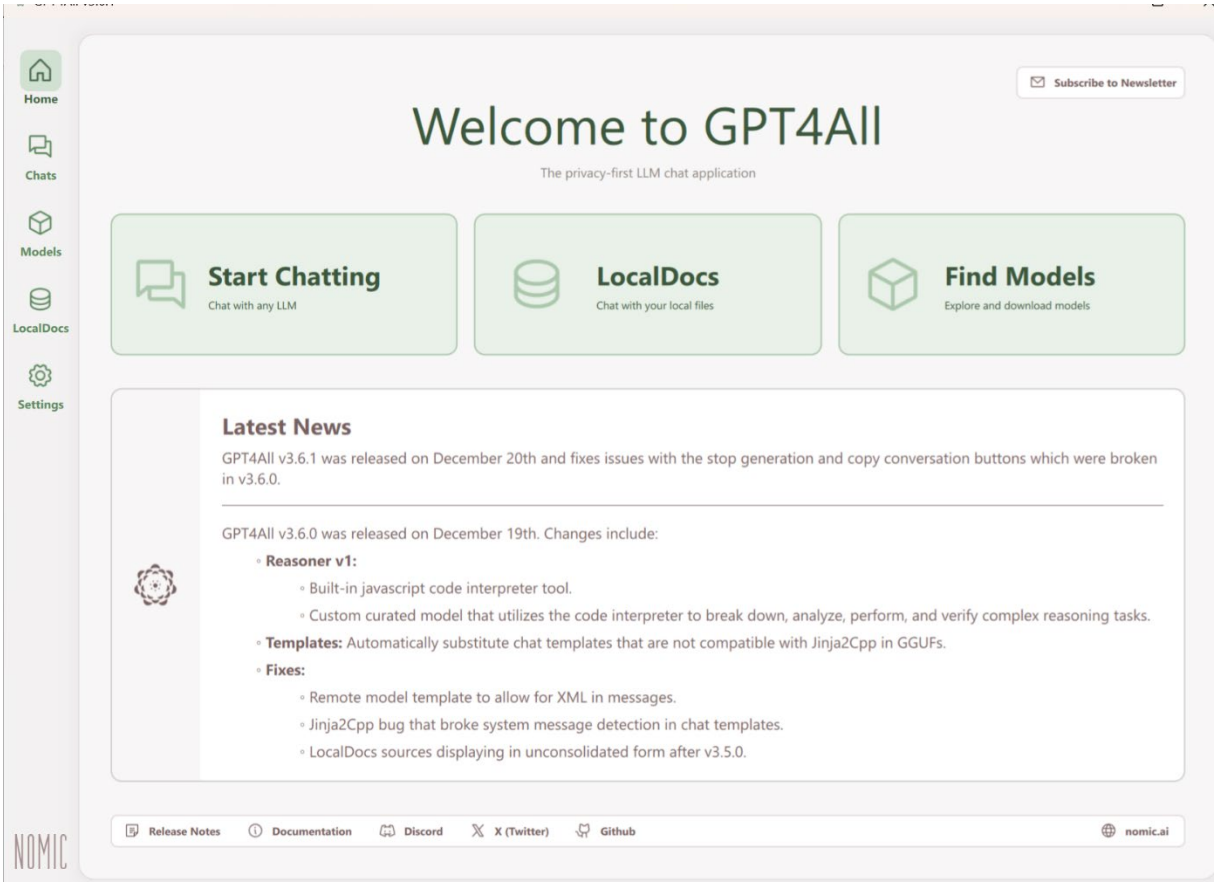

Abbildung 3 GPT4All Programminterface

Llama 3 Modell herunterladen

1. Öffnen Sie GPT4All und navigieren Sie zum Tab **"Find Models"**.
2. Suchen Sie nach dem **Llama 3 Instruct** Modell und laden Sie es herunter.

Erste Schritte mit GPT4All

Starten eines Chats

1. Klicken Sie auf **"Start Chatting"** auf der Startseite.
2. Wählen Sie das heruntergeladene **Llama 3 Modell** aus.

Grundlegende Interaktion

- Beginnen Sie, Fragen zu stellen oder Anweisungen zu geben.
- Das Modell generiert daraufhin passende Antworten.

Einrichtung von LocalDocs für RAG

Erstellen einer Dokumentensammlung

1. Klicken Sie auf **"+ Add Collection"** in der GPT4All-Oberfläche.
2. Benennen Sie Ihre Sammlung und verknüpfen Sie sie mit einem Ordner, der Ihre Dokumente enthält.

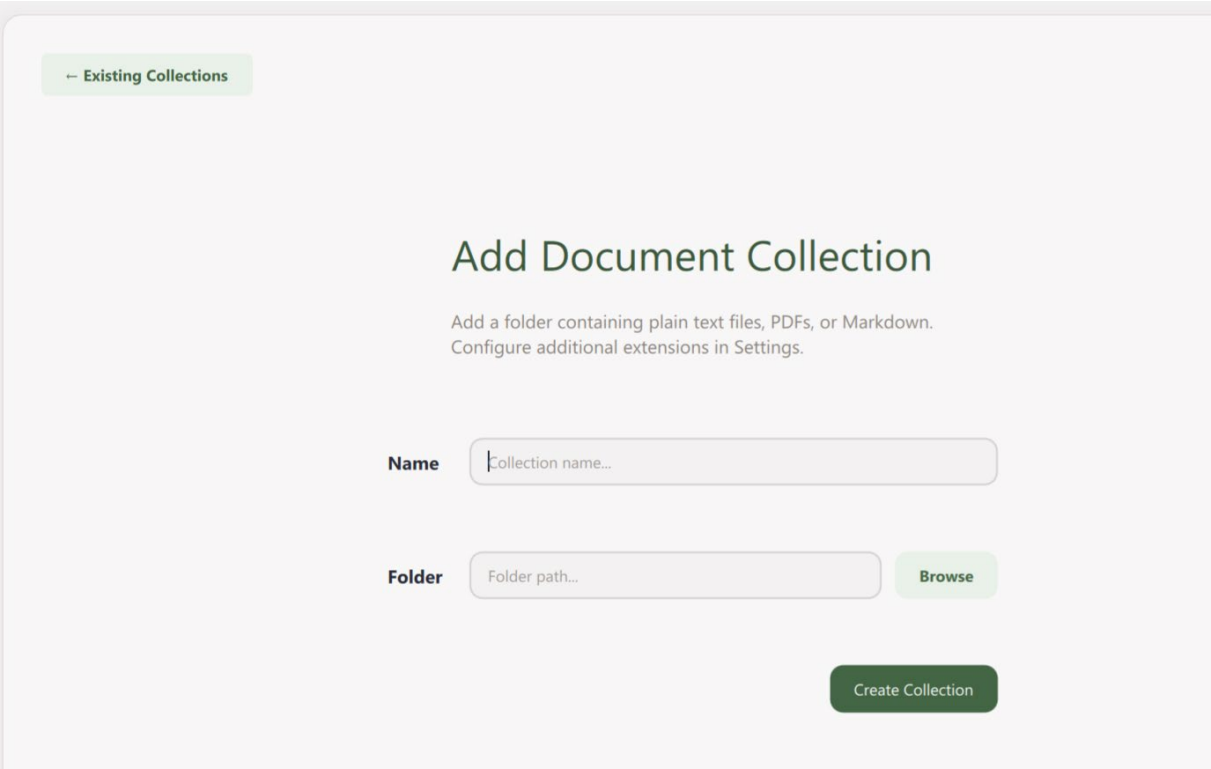

Abbildung 4 GPT4ALL Document Collection

Dokumente verarbeiten

- GPT4All verarbeitet und indexiert Ihre Dokumente automatisch.
- Warten Sie, bis der Prozess abgeschlossen ist. Ein grüner **"Ready"-Indikator** signalisiert die erfolgreiche Verarbeitung.

LocalDocs aktivieren

1. In der Chat-Oberfläche finden Sie einen **LocalDocs-Button** in der oberen rechten Ecke.
2. Aktivieren Sie diesen, um Ihre lokale Wissensbasis in den Chat einzubeziehen.

Nutzung von RAG mit LocalDocs

Stellen von Fragen zu Ihren Dokumenten

- Formulieren Sie gezielte Fragen zu Ihren Dokumenten.
- GPT4All extrahiert relevante Informationen und integriert sie in die Antwort.

Quellenangaben prüfen

- GPT4All zeigt an, aus welchen Dokumenten die bereitgestellten Informationen stammen.
- Dies erleichtert die Nachvollziehbarkeit und Validierung.

Optimierung der Ergebnisse

Präzisere Antworten erhalten

- Verwenden Sie spezifische Begriffe aus Ihren Dokumenten.
- Experimentieren Sie mit verschiedenen Frageformulierungen.

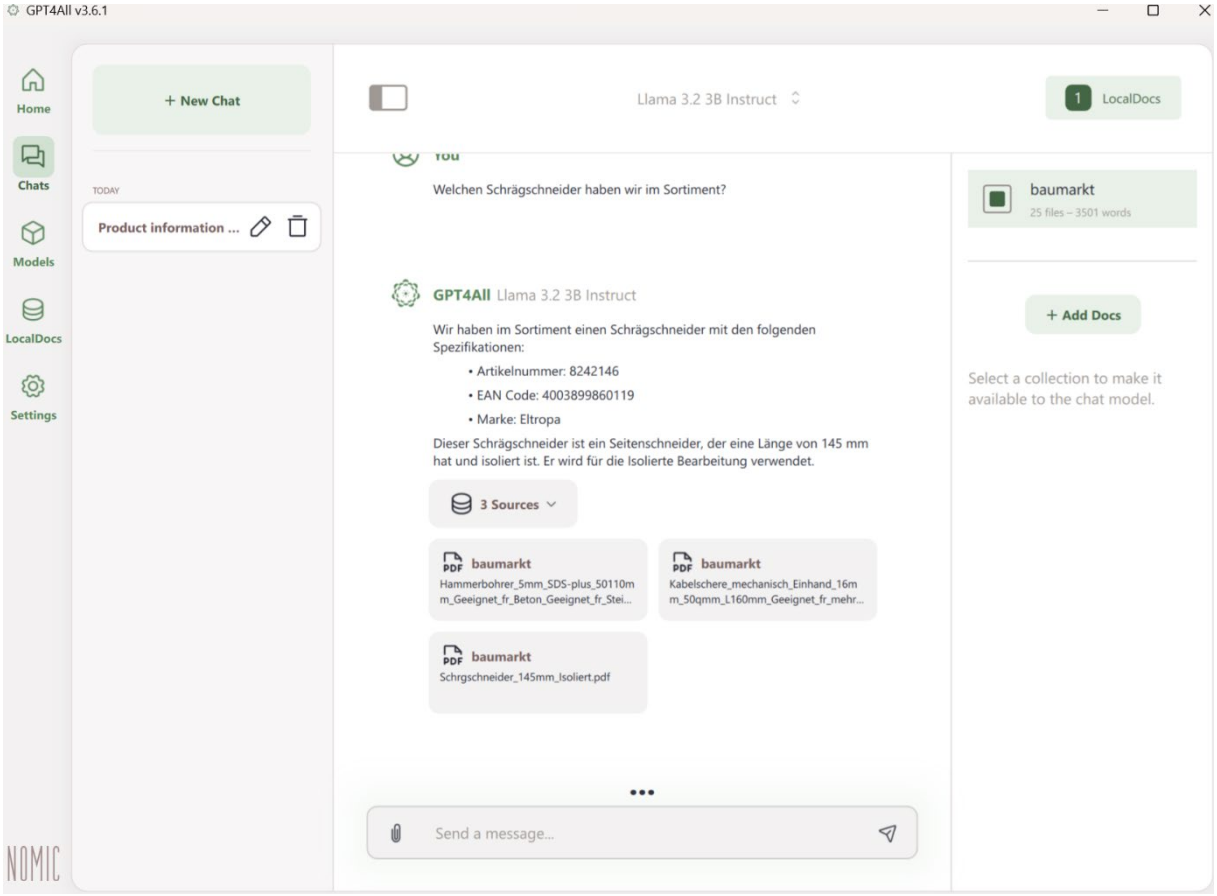

Abbildung 5 RAG via Llama3.2 3B

Fortgeschrittene Einstellungen

Zugriff auf erweiterte Optionen

1. Klicken Sie auf das **Zahnrad-Symbol** auf der Startseite.
2. Passen Sie verschiedene Parameter an, um das Verhalten des Modells zu optimieren.

Anpassung der Modellparameter

- Experimentieren Sie mit Einstellungen wie **Kontextlänge** und **Temperatur**, um die Qualität und Kreativität der Antworten zu beeinflussen.

GPU-Beschleunigung aktivieren

- Wenn Ihr System über eine kompatible GPU verfügt, aktivieren Sie die GPU-Beschleunigung für schnellere Inferenzzeiten.

Eigene Dokumente mit AnythingLLM und Ollama nutzen

AnythingLLM ist ein leistungsstarkes Tool, das Ihnen ermöglicht, Ihre eigenen Dokumente mit Hilfe von Ollama als LLM-Provider lokal zu analysieren und Fragen dazu zu stellen. Diese Anleitung führt Sie durch die Einrichtung und Nutzung von AnythingLLM mit dem Fokus auf Retrieval-Augmented Generation (RAG).

Installation und Einrichtung

Voraussetzungen

1. Installieren Sie AnythingLLM auf Ihrem Computer.
2. Stellen Sie sicher, dass Ollama bereits installiert und konfiguriert ist.

AnythingLLM herunterladen und installieren

1. Besuchen Sie die offizielle AnythingLLM-Website.
2. Laden Sie die Version herunter, die zu Ihrem Betriebssystem passt (Windows, Mac oder Linux).
3. Folgen Sie den Installationsanweisungen und stellen Sie sicher, dass alle Abhängigkeiten erfüllt sind.

Einrichtung von AnythingLLM

Schritt 1: AnythingLLM konfigurieren

1. Öffnen Sie AnythingLLM.
2. Navigieren Sie zu den **Einstellungen**.
3. Wählen Sie unter **"LLM Preferences"** Ollama als LLM-Provider aus.
4. Geben Sie als Base URL http://127.0.0.1:11434 ein (dies ist die lokale Ollama-Serveradresse).
5. Setzen Sie den Kontextwert auf 4096.
6. Speichern Sie die Einstellungen.

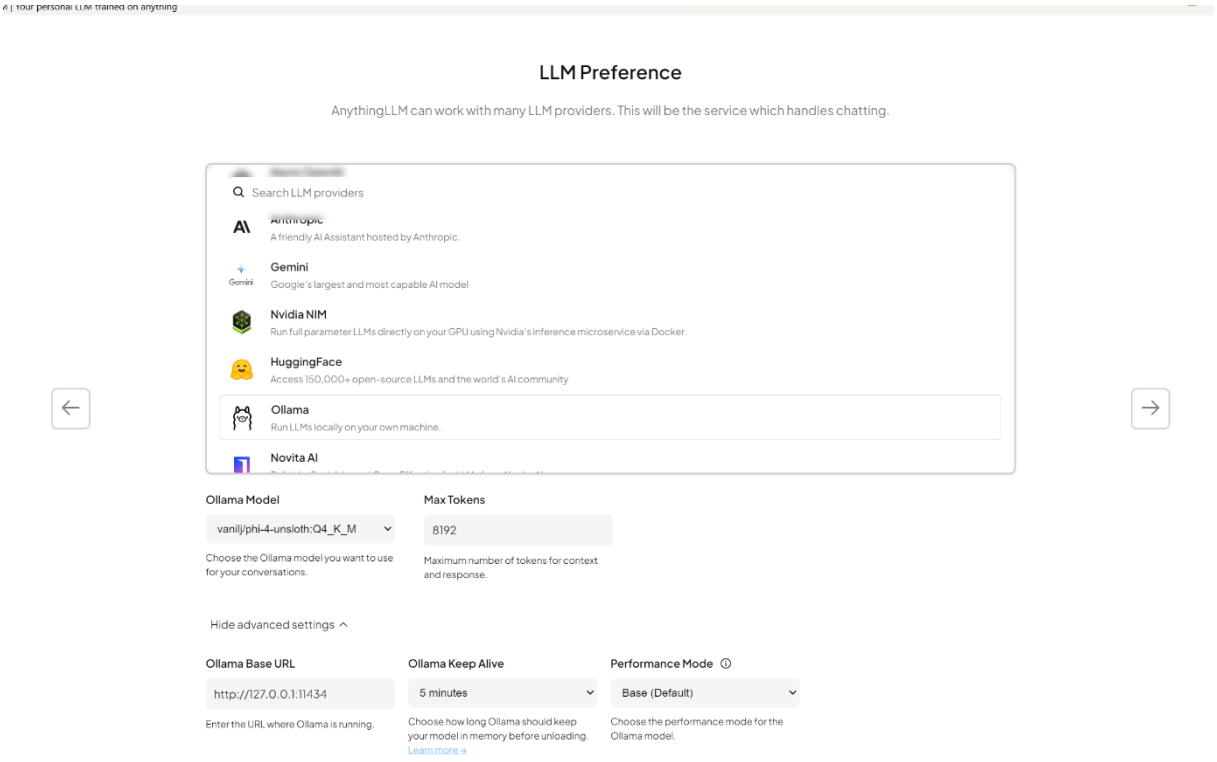

Abbildung 6 AnythingLLM Erst-Einrichtung

Schritt 2: Einen Workspace erstellen

1. Kehren Sie zur Hauptoberfläche von AnythingLLM zurück.
2. Klicken Sie auf **"New Workspace"**.
3. Vergeben Sie einen Namen für Ihren Workspace, z. B. "Meine Dokumente".

Schritt 3: Dokumente hochladen

1. Öffnen Sie Ihren erstellten Workspace.
2. Klicken Sie auf **"Upload"** oder das Upload-Symbol.
3. Wählen Sie die gewünschten PDF-Dokumente aus und laden Sie sie hoch.
4. Warten Sie, bis die Verarbeitung abgeschlossen ist und die Dokumente indexiert wurden.

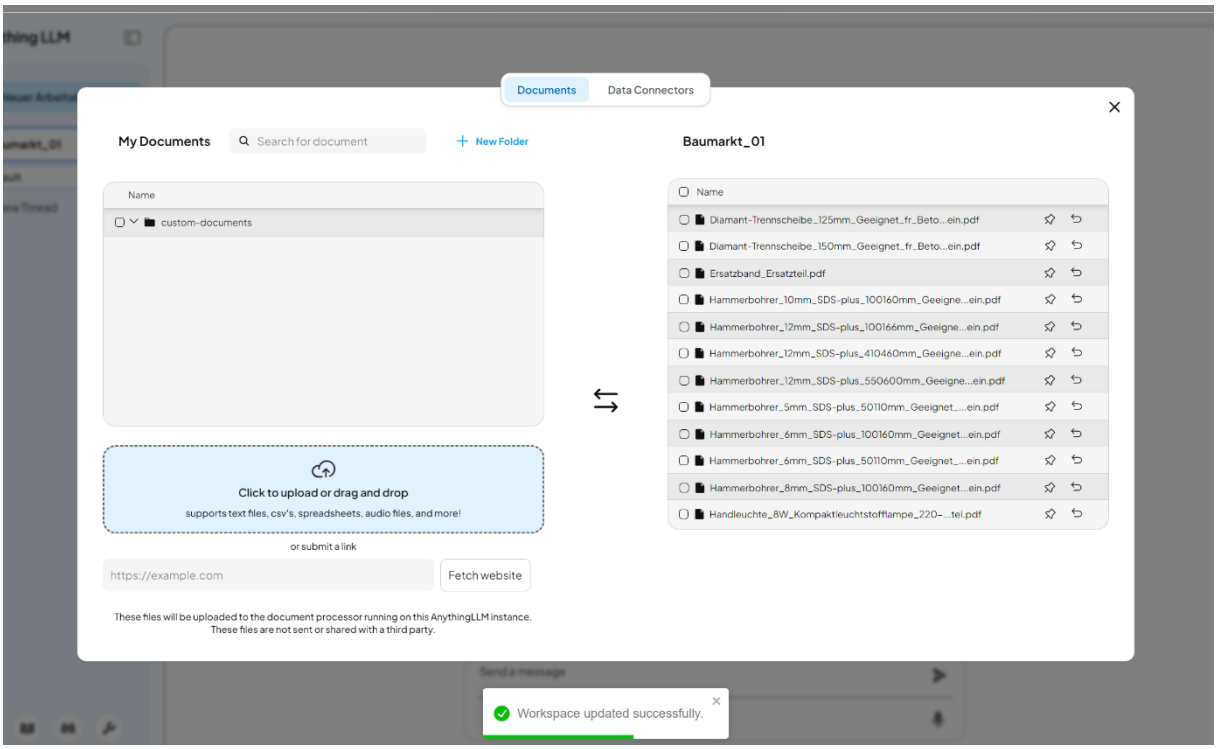

Abbildung 7 Dokumente hochladen

Mit Ihren Dokumenten interagieren

Fragen zu Ihren Dokumenten stellen

1. Sobald Ihre Dokumente verarbeitet wurden, können Sie in das Chatfeld Fragen eingeben.
2. AnythingLLM wird relevante Informationen aus den hochgeladenen Dokumenten extrahieren und eine Antwort generieren.

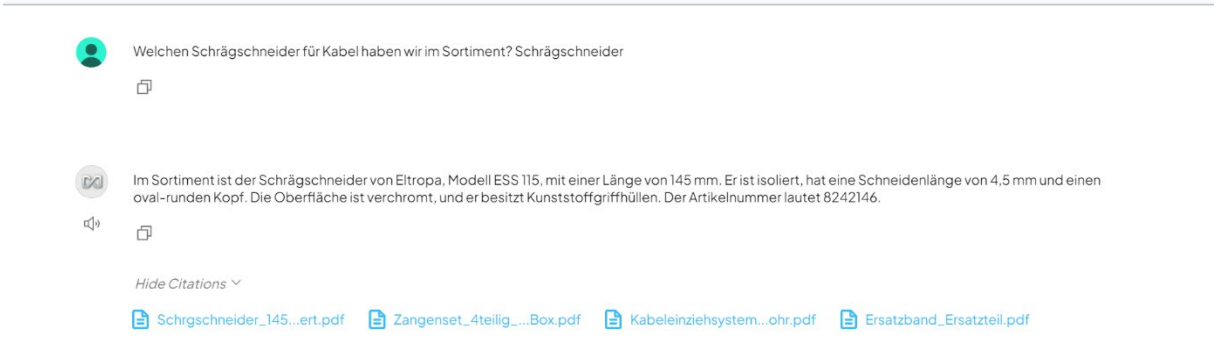

Abbildung 8 RAG via Phi4 in Anything LLM

Optimierung der Interaktion

- Formulieren Sie präzise Fragen, die sich direkt auf den Dokumenteninhalt beziehen.

- Experimentieren Sie mit unterschiedlichen Ollama-Modellen, um die Qualität der Antworten zu verbessern.
- Nutzen Sie die Möglichkeit, mehrere Dokumente in einem Workspace zu kombinieren, um umfassendere Antworten zu erhalten.

Tipps für optimale Ergebnisse

Eine gute Strukturierung und Lesbarkeit Ihrer Dokumente verbessert die Extraktion relevanter Informationen erheblich. Aktualisieren Sie regelmäßig Ihre Dokumentensammlung, um immer auf dem neuesten Stand zu bleiben. Probieren Sie verschiedene Modelle aus, um herauszufinden, welches am besten zu Ihren Anforderungen passt.

Modellvergleiche mit LM Arena

In der Regel sind normale Office-PCs ohne dedizierte Grafikkarte nicht leistungsstark genug, um große Modelle wie LLama 3 70B oder gar 405b lokal auszuführen. Dennoch gibt es Möglichkeiten, diese Modelle kostenlos zu testen und ihre Leistungsfähigkeit zu bewerten. LM Arena bietet die Möglichkeit, mehrere Modelle zu testen und direkt miteinander zu vergleichen. Nutzer können verschiedene Modelle in interaktiven Szenarien ausprobieren und so fundierte Entscheidungen darüber treffen, welches Modell am besten für ihre Anforderungen geeignet ist.

Fazit und Ausblick

Der Einsatz lokaler LLMs stellt eine attraktive Alternative zu cloudbasierten Lösungen dar. Sie ermöglichen eine höhere Datensicherheit, Unabhängigkeit von Internetverbindungen und langfristige Kosteneinsparungen. Mit den richtigen Tools und etwas technischer Vorbereitung können sowohl Unternehmen als auch Privatpersonen von den Vorteilen profitieren. In Zukunft ist mit weiteren Fortschritten in der Modelloptimierung und Hardwareeffizienz zu rechnen, wodurch der Einsatz lokaler KI-Modelle noch attraktiver wird.

Die vorgestellten Tools – Ollama, GPT4All und AnythingLLM – bieten unterschiedliche, aber komplementäre Lösungen für die Nutzung lokaler Sprachmodelle. Mit Ollama lassen sich verschiedenste Open-Source-Modelle einfach auf dem eigenen Rechner ausführen, was eine flexible und skalierbare Nutzung ermöglicht. GPT4All bietet eine benutzerfreundliche Möglichkeit, große Sprachmodelle lokal auszuführen und dabei

eigene Dokumente über die RAG-Funktion (Retrieval-Augmented Generation) zu integrieren, wodurch wertvolle Informationen aus persönlichen Daten gewonnen werden können. AnythingLLM schließlich vereinfacht die Verwaltung und Nutzung eigener Dokumente durch eine strukturierte Oberfläche und die Möglichkeit, mit Ollama als Backend-Provider zu arbeiten, um gezielt Antworten aus den eigenen Daten zu erhalten.