# Modellbeschreibung

## Quellen (Beispiele in den Tabellen stammen von den offiziellen Produkt-Ankündigungen)

- OpenAI – Introducing GPT-4.1 (Details zu 4.1 / mini / nano und Benchmarks).
- OpenAI – Introducing GPT-5 (Beispiele: Thinking, Router, Einsatzszenarien).
- OpenAI – Introducing o3 and o4-mini (Reasoning, Tool-Use, AIME / Benchmarks).

## GPT-4.1-Serie vs. GPT-5-Serie

| **Modell** | **Kerneigenschaft (in einem Satz)** | **Konkretes Alltags-Beispiel (einfach verständlich)** |
|---|---|---|
| GPT-4.1 | Starker Allrounder für Coding, lange Kontexte (bis zu 1M Tokens), gute Instruktions-Folge | „Suche in einem ganzen Buch nach allen Erwähnungen eines Fachbegriffs und fasse Kapitel 3 zusammen“ — macht das zuverlässig. |
| GPT-4.1-mini | Kompakter, schneller, günstigere API-Option für viele Aufgaben | „Automatisch E-Mails klassifizieren und Schlagwörter vorschlagen“ — schnell + günstig. |
| GPT-4.1-nano | Sehr klein und extrem schnell; ideal für einfache Klassifikation/Autovervollständigung | „Kurzantworten oder Kategorie-Tags für Support-Tickets“ — viel günstiger pro Anfrage. |
| GPT-5 | Größere „Denktiefe“ und Router/Thinking-Modi; besser bei komplexen Analysen, Multimodalität | „Mehrstufiges Recherche-Projekt: Web-Abfragen + Code zum Auswerten von Daten + Ergebnisbericht“ — kann Tools/Plugins nutzen und länger denken. |
| GPT-5-mini | Mini-Variante von GPT-5 — schneller, für viele Nutzer-Anfragen, behält wichtige Verbesserungen | „Schnelle interne Zusammenfassungen großer Dokumente für Team-Chats“. |
| GPT-5-nano | Nano-Variante für sehr hohes Volumen / einfache Tasks; günstigste GPT-5-Option | „Automatische Stichwort-Extraktion in hohen Volumina (Logs, Chats)“. |

## o-Serie: o3 und o4-mini


Dieses Dokument stammt aus der Dokuwiki-Instanz mit dem Titel RZ-Dokumentationen, die vom Rechenzentrum der Technischen Universität Clausthal bereitgestellt wird.

| Modell | Typischer Einsatz (Warum man es wählt) | Konkretes Beispiel |
|---|---|---|
| o3 | Hoche Leistung beim logischen, multimodalen Problemlösen; starkes Reasoning, gut für Forschung & Programmier-Debugging | „Analysiere ein technisches Paper, teste Hypothesen mit Python-Schnipseln und generiere reproduzierbare Plots“ — gut für Forschungsteams. |
| o4-mini | Optimierte, kosteneffiziente Reasoning-Variante; sehr schnell und gut bei Mathe/Coding mit Toolzugriff | „Löse viele Mathe-Aufgaben (AIME-ähnlich) unter Verwendung eines eingebetteten Python-Interpreters“ — sehr gutes Preis/Leistungsergebnis. |

Direkt-Link:
**https://doku.tu-clausthal.de/doku.php?id=sonstige_dienste:ki-dienste:librechat:modelle_unberschied&rev=1755682460**

Letzte Aktualisierung: **09:34 20. August 2025**


Dieses Dokument stammt aus der Dokuwiki-Instanz mit dem Titel RZ-Dokumentationen, die vom Rechenzentrum der Technischen Universität Clausthal bereitgestellt wird.