M17a - Modellauswahl

Anwendung Generativer KI

Stand: 04.2025

Die moderne KI-Landschaft entwickelt sich rasant und bietet eine Vielzahl spezialisierter Modelle für unterschiedliche Anwendungsfälle. Dieses Dokument bietet eine umfassende Übersicht zu Modelltypen, Auswahlkriterien und Bewertungsmethoden, um fundierte Entscheidungen bei der Implementierung von KI-Lösungen zu ermöglichen.

1 KI-Modelllandschaft

Die Vielfalt moderner KI-Modelle ermöglicht maßgeschneiderte Lösungen für spezifische Aufgabenstellungen. Von spezialisierten Reasoning-Modellen bis hin zu multimodalen Systemen bietet die Technologielandschaft unterschiedliche Optionen, deren Einsatz ein fundiertes Verständnis ihrer jeweiligen Stärken und Limitationen erfordert.

Die moderne KI-Landschaft bietet verschiedene spezialisierte Modelltypen für unterschiedliche Anwendungsfälle:

- **Reasoning-Modelle**: Spezialisiert auf logisches Denken und Problemlösung (z.B. o3-mini)
- **Sprachmodelle**: Für natürlichsprachliche Aufgaben (z.B. GPT-4)
- **Codex-Modelle**: Für Codegenerierung und Programmieraufgaben
- **Bildgenerierungsmodelle**: Erzeugen Bilder aus textlichen Beschreibungen (z.B. DALL-E)

- **Sprachverarbeitungsmodelle**: Für Spracherkennung und -transkription (z.B. Whisper)

2 Modellkategorisierung

Um die Vielzahl verfügbarer KI-Modelle übersichtlich zu strukturieren, werden sie häufig in Tiers kategorisiert. Diese Einteilung berücksichtigt Faktoren wie Leistungsfähigkeit, Anwendungsbereich und Ressourcenbedarf. Eine solche Kategorisierung hilft bei der Vorauswahl geeigneter Modelle für spezifische Anforderungen und ermöglicht eine effiziente Ressourcenallokation.

Tier	Beispiele	Eigenschaften	Anwendungsfälle
Tier 3	GPT-4o mini, Gemini 1.5 Flash, Claude 3 Haiku, o3-mini	Preiswert, schnell, ressourceneffizient	Datenzusammenfassung, schnelle Textgenerierung, lokale Verarbeitung
Tier 2	GPT-4 Turbo, Claude 3.7 Sonnet, Gemini 2.0 Pro	Gutes Preis-Leistungs-Verhältnis, vielseitig einsetzbar	Programmierung, Funktionsaufrufe, Tool-Nutzung
Tier 1	GPT-4o, Claude 3.5 Opus, Gemini 1.0 Ultra	Höchste Intelligenz, fortschrittliche Reasoning-Fähigkeiten	Komplexe Reasoning-Aufgaben, tiefes Verständnis

3 Modellauswahlprozess

Die Auswahl des optimalen KI-Modells für eine spezifische Anwendung erfordert einen strukturierten Prozess. Dieser systematische Ansatz hilft dabei, die Anforderungen klar zu definieren, relevante Modelle zu identifizieren und eine fundierte Entscheidung zu treffen. Der folgende mehrstufige Prozess bietet eine Orientierung, um das am besten geeignete Modell auszuwählen und erfolgreich zu implementieren.

3.1 Anforderungsanalyse

- Definition der Aufgaben (Textgenerierung, Fragebeantwortung, etc.)
- Festlegung von Qualitätskriterien (Kohärenz, Genauigkeit)
- Identifikation notwendiger Domänenkenntnisse

- Anforderungen an Antwortgeschwindigkeit
- Budget-Rahmenbedingungen

3.2 Bewertungskriterien

- Verständlichkeit der Ausgaben
- Effizienz und Geschwindigkeit
- Skalierbarkeit
- Kosten

3.3 Recherche und Vorauswahl

- Analyse bestehender Modelle anhand festgelegter Kriterien

3.4 Praktische Modellbewertung

- Quantitative Methoden (Benchmarking, Metriken)
- Qualitative Verfahren (Nutzerrückmeldungen)
- Testphase zur praktischen Erprobung

3.5 Finale Auswahl und Implementierung

- Entscheidung für das am besten geeignete Modell
- Integration und kontinuierliches Monitoring

4 Modellkaskade: Kombination von Modellen

Die Modellkaskade stellt einen innovativen Ansatz dar, bei dem mehrere KI-Modelle in einer sorgfältig orchestrierten Abfolge zusammenarbeiten. Dieses Konzept ermöglicht es, die spezifischen Stärken verschiedener Modelle zu kombinieren und dabei ihre

jeweiligen Schwächen zu kompensieren. Durch die gezielte Zuweisung von Teilaufgaben an die jeweils am besten geeigneten Modelle lassen sich komplexe Problemstellungen effizienter, kostengünstiger und mit höherer Qualität lösen als mit einem einzelnen Modell.

Die Modellkaskade kombiniert verschiedene Modelle entsprechend ihrer Stärken für spezifische Teilaufgaben. Ein Beispiel für die Erstellung eines wissenschaftlichen Berichts:

1. **Datenanalyse mit pandas**: Analyse großer Datensätze, statistische Zusammenfassungen
2. **Logische Strukturierung mit o3-mini**: Strukturierung der Ergebnisse, Erstellung einer logischen Gliederung
3. **Kreative Textgenerierung mit GPT-4o**: Verfassen des Berichts in ansprechender Sprache
4. **Multimodale Präsentation mit gpt-4.5 und plotly-express**: Ergänzung durch visuelle Elemente

4.1 Code-Beispiel für eine Modellkaskade

```
import pandas as pd
from openai import OpenAI
import plotly.express as px

# OpenAI-Client initialisieren
client = OpenAI()

# Schritt 1: Datenanalyse mit pandas
def daten_analyse(datei_pfad):
    """
    Führt eine grundlegende Datenanalyse durch, einschließlich statistischer Zusammenfassung
    und Korrelationsmatrix.
    """
    # Beispiel-Datensatz laden
    daten = pd.read_csv(datei_pfad)

    # Statistische Analyse durchführen
    zusammenfassung = daten.describe()
    korrelationen = daten.corr()
```

```
    return daten, zusammenfassung, korrelationen

# Schritt 2: Logische Strukturierung mit o3-mini
def logische_strukturierung(zusammenfassung, korrelationen):
    """
    Erstellt eine logische Gliederung für einen wissenschaftlichen Bericht.
    """
    prompt = f"""
    Erstelle eine logische Gliederung für einen wissenschaftlichen Bericht basierend auf den Daten:
    Statistische Zusammenfassung: {zusammenfassung}
    Korrelationsmatrix: {korrelationen}
    """

    response = client.chat.completions.create(
        model="o3-mini",
        messages=[
            {"role": "system", "content": "Du bist ein hilfreicher Assistent, der wissenschaftliche Berichte
strukturiert."},
            {"role": "user", "content": prompt}
        ],
        max_tokens=500
    )

    gliederung = response.choices[0].message.content
    return gliederung

# Schritt 3: Kreative Textgenerierung mit GPT-4o
def kreative_textgenerierung(gliederung):
    """
    Generiert einen wissenschaftlichen Bericht basierend auf der Gliederung.
    """
    prompt = f"Schreibe einen wissenschaftlichen Bericht basierend auf: {gliederung}"
```

```
    response = client.chat.completions.create(
        model="gpt-4o",
        messages=[
            {"role": "system", "content": "Du bist ein hilfreicher Assistent, der wissenschaftliche Berichte
schreibt."},
            {"role": "user", "content": prompt}
        ],
        max_tokens=2000
    )

    bericht = response.choices[0].message.content
    return bericht

# Schritt 4: Multimodale Präsentation
def multimodale_praesentation(daten, client):
    """
    Erstellt Visualisierungen der Daten mit Plotly Express.
    """
    # Numerische Spalten identifizieren
    numerische_spalten = daten.select_dtypes(include=['number']).columns.tolist()

    if len(numerische_spalten) < 2:
        raise ValueError("Nicht genügend numerische Spalten für Visualisierung")

    # Einfache Datenbeschreibung erstellen
    daten_beschreibung = daten[numerische_spalten].describe().to_dict()
    korrelationen = daten[numerische_spalten].corr().to_dict()

    # GPT-4o für Visualisierungsvorschläge nutzen
    prompt = f"""
    Analysiere die Daten und empfiehl eine Visualisierung im JSON-Format:
    Spalten: {numerische_spalten}
    Statistiken: {daten_beschreibung}
    Korrelationen: {korrelationen}
```

```
    """

    response = client.chat.completions.create(
        model="gpt-4o",
        messages=[
            {"role": "system", "content": "Du bist ein Datenvisualisierungsexperte. Antworte im JSON-Format."},
            {"role": "user", "content": prompt}
        ],
        response_format={"type": "json_object"}
    )

    import json
    empfehlung = json.loads(response.choices[0].message.content)

    # Visualisierung erstellen
    fig = px.scatter(daten, x=numerische_spalten[0], y=numerische_spalten[1],
                     title=empfehlung.get("titel", "Datenpunkte"))

    return fig

# Hauptprogramm
if __name__ == "__main__":
    datei_pfad = "daten.csv"
    daten, zusammenfassung, korrelationen = daten_analyse(datei_pfad)
    gliederung = logische_strukturierung(zusammenfassung, korrelationen)
    bericht = kreative_textgenerierung(gliederung)
    fig = multimodale_praesentation(daten, client)
```

4.2 Vorteile einer Modellkaskade

1. **Effizienzsteigerung**: Jedes Modell wird für seine Stärken eingesetzt
2. **Kostenoptimierung**: Ressourcenschonende Modelle für einfache Aufgaben

3. **Flexibilität**: Bearbeitung unterschiedlichster Anforderungen

5 Bewertungsmethoden für KI-Modelle

Die systematische Bewertung von KI-Modellen ist entscheidend, um ihre Leistungsfähigkeit, Eignung und Grenzen zu verstehen. Umfassende Evaluierungsmethoden ermöglichen nicht nur einen objektiven Vergleich zwischen verschiedenen Modellen, sondern unterstützen auch die kontinuierliche Verbesserung und Weiterentwicklung der Technologie. Die folgenden Abschnitte stellen wichtige Benchmarks, Bewertungsdimensionen und Metriken vor, die für eine fundierte Beurteilung von KI-Modellen unerlässlich sind.

5.1 Benchmarks für Sprachmodelle

- **MMLU (Massive Multitask Language Understanding)**: Standard-Benchmark über 57 Fachgebiete

Modell	MMLU-Score
GPT-4o	88,7%
Gemini 2.0 Ultra	90,0%
Claude 3 Opus	88,2%
Llama 3.1 405B	87,3%
gpt-4o-mini	70,0%

5.2 Bewertungsdimensionen

Die Bewertung von LLMs umfasst verschiedene Dimensionen:

1. **Wissens- und Fähigkeitsbewertung**:
   - Beantwortung von Fragen
   - Wissensvervollständigung
   - Logisches und mathematisches Denken

    - Werkzeugnutzung
2. **Alignment-Bewertung**:
    - Übereinstimmung mit menschlichen Werten
    - Ethische und moralische Aspekte
    - Bias und Fairness
    - Toxizität und Sicherheit
    - Wahrhaftigkeit der Antworten
3. **Sicherheitsbewertung**:
    - Robustheit gegenüber Störungen und Angriffen
    - Analyse potenzieller Risiken

5.3 Bewertungsmethoden und Metriken

5.3.1 Automatisierte Metriken

- **BLEU (Bilingual Evaluation Understudy)**: Misst N-Gramm-Überschneidungen zwischen generiertem und Referenztext
- **ROUGE (Recall-Oriented Understudy for Gisting Evaluation)**: Bewertet die Qualität von Zusammenfassungen

5.3.2 Humanbasierte Bewertung

- Grammatikalität, Kohäsion, Gefälligkeit, Relevanz
- Elo-Bewertungssystem für direkten Modellvergleich

5.3.3 LLMs als Evaluatoren

- Einsatz leistungsfähiger Modelle zur Bewertung von Textqualität
- Black-Box-Halluzinationserkennung (z.B. SelfCheckGPT)

5.3.4 Datensätze zur Bewertung

- **Frage-Antwort**: SQuAD, NarrativeQA, HotpotQA, CoQA
- **Common Sense Reasoning**: ARC, QASC, HellaSWAG, PIQA
- **Logisches Denken**: SNLI, MultiNLI, ReClor
- **Mathematisches Denken**: GSM8K, MATH, CMATH
- **Ethik und Moral**: Social Chemistry 101
- **Bias und Toxizität**: OLID, HateXplain, RealToxicityPrompts
- **Wahrhaftigkeit**: SelfAware, DIALFACT
- **Robustheit**: AdvGLUE, ANLI

5.4 Herausforderungen in der Bewertung

- **Unklare Bewertungskriterien**: Subjektivität bei bestimmten Aspekten wie Grammatikalität
- **Bias in LLMs und Bewertungsdatensätzen**: Notwendigkeit zur Identifikation und Reduzierung
- **Robustheit**: Anfälligkeit für adversarial attacks und Prompt-Variationen
- **Bewertung fortgeschrittener Fähigkeiten**: Machtstreben, Situationsbewusstsein
- **Tokenisierungseffekte**: Einfluss auf Effizienz und Robustheit der Modelle

6 Praktische Anwendungen und Aufgaben

Die theoretischen Grundlagen zur Modellauswahl und -bewertung finden ihre praktische Anwendung in konkreten Projekten und Aufgabenstellungen. Dieser Abschnitt bietet eine Orientierung für die Umsetzung der vorgestellten Konzepte in der Praxis. Von der detaillierten Anforderungsanalyse über den Modellvergleich bis hin zur qualitativen Evaluation werden verschiedene Anwendungsszenarien vorgestellt, die als Ausgangspunkt für eigene KI-Projekte dienen können.

6.1 Anforderungsanalyse für ein KI-Projekt

- Definition primärer Funktionen
- Spezifische Anforderungen an das Sprachverständnis
- Notwendige Fachkenntnisse

- Anforderungen an Antwortgeschwindigkeit
- Budget-Rahmenbedingungen

6.2 Vergleichsanalyse bekannter KI-Modelle

- Leistungsmerkmale (MMLU-Score, Kontextfenstergröße)
- Antwortlatenz und Kosten
- Verfügbarkeit und unterstützte Sprachen
- Multimodale Fähigkeiten

6.3 Qualitative Evaluation eines Sprachmodells

- Bewertungsschema mit 5-7 qualitativen Kategorien
- Testfragen und Bewertungskriterien
- Evaluationsprozess und Vermeidung von Bewertungsverzerrungen
- Kombination qualitativer und quantitativer Metriken

Fazit

Zusammenfassend lässt sich sagen, dass die **Evaluierung von Large Language Models (LLMs) ein wichtiges Forschungsgebiet** ist, um ihre Fähigkeiten und Grenzen zu verstehen. Die Evaluierung umfasst verschiedene **Attribute wie Grammatikalität, Kohäsion, Gefallen, Relevanz, Flüssigkeit und Bedeutungserhalt**. Sowohl **menschliche Evaluatoren als auch LLMs selbst werden zur Bewertung eingesetzt**. Es gibt **spezifische Benchmarks und Datensätze** zur Bewertung von LLMs in verschiedenen Bereichen wie **Textgenerierung, Fragebeantwortung und Zusammenfassung**.

Ein wichtiger Aspekt der LLM-Evaluierung ist die **Sicherheitsbewertung**, die **Robustheit gegenüber adversarialen Angriffen** (manipulierte Eingaben, um LLM in die Irre zu führen) und die Identifizierung von **Risiken wie Bias und Toxizität** umfasst. Die Evaluierung kann auch auf **spezialisierte LLMs** in Bereichen wie Medizin, Recht und Finanzen zugeschnitten sein.

Verschiedene **Metriken, darunter Likert-Skalen und der BLEU-Score**, werden zur Quantifizierung der LLM-Leistung verwendet. Es gibt auch **Tools und Frameworks wie DeepEval**, die die Evaluierung erleichtern. Es ist wichtig zu beachten, dass **Evaluierungsbias existieren können**, beispielsweise eine Präferenz für längere Texte. Die **ethischen Aspekte** spielen ebenfalls eine Rolle bei der Entwicklung und Nutzung von LLMs.

7 A | Aufgabe

Die Aufgabestellungen unten bieten Anregungen, Sie können aber auch gerne eine andere Herausforderung angehen.

Anforderungsanalyse für ein KI-Projekt

Entwickeln Sie eine strukturierte Anforderungsanalyse für ein fiktives oder reales KI-Projekt.

Aufgabenstellung:

1. Wählen Sie einen konkreten Anwendungsfall (z.B. Kundenservice-Chatbot für eine Bank, Content-Generator für Social Media, oder Übersetzungstool für technische Dokumentation).
2. Definieren Sie:
   - Die primären Funktionen, die das KI-Modell erfüllen soll
   - Die spezifischen Anforderungen an das Sprachverständnis
   - Notwendige Fachkenntnisse in relevanten Domänen
   - Anforderungen an die Antwortgeschwindigkeit
   - Budget-Rahmenbedingungen
3. Erstellen Sie eine Prioritätenliste dieser Anforderungen (unbedingt erforderlich, wichtig, wünschenswert).
4. Beschreiben Sie, welche Kompromisse Sie bei konkurrierenden Anforderungen eingehen würden.

Abgabeformat:
Erstellen Sie ein Dokument mit Ihrer Anforderungsanalyse (1-2 Seiten).

Vergleichsanalyse bekannter KI-Modelle

Führen Sie eine vergleichende Analyse von mindestens drei verschiedenen KI-Modellen anhand vorgegebener Bewertungskriterien durch.

Aufgabenstellung:

1. Wählen Sie drei KI-Modelle aus der folgenden Liste aus:
   - GPT-4o
   - Claude 3 Opus
   - Gemini 2.0 Ultra
   - Llama 3.1
   - Mistral 7B
   - Ein anderes aktuelles KI-Modell Ihrer Wahl
2. Recherchieren Sie die Leistungsmerkmale dieser Modelle anhand der folgenden Kriterien:
   - MMLU-Score oder vergleichbare Benchmark-Ergebnisse
   - Kontextfenstergröße
   - Antwortlatenz
   - Kosten (pro Token oder alternativer Maßstab)
   - Verfügbarkeit (API, Open-Source, etc.)
   - Unterstützte Sprachen
   - Multimodale Fähigkeiten (falls vorhanden)
3. Erstellen Sie eine Bewertungstabelle mit den recherchierten Informationen.
4. Verfassen Sie eine begründete Empfehlung, welches dieser Modelle sich für folgende Szenarien am besten eignen würde:
   - Entwicklung eines kostengünstigen Chatbots für ein kleines Unternehmen
   - Erstellung von KI-generierten Inhalten für ein internationales Nachrichtenportal

- Unterstützung bei der Software-Entwicklung

Abgabeformat:
Vergleichstabelle mit Bewertungen und einer Seite mit Ihren Empfehlungen.

Konzept für die qualitative Evaluation eines Sprachmodells

Entwickeln Sie ein strukturiertes Testverfahren zur qualitativen Bewertung eines Sprachmodells.

Aufgabenstellung:

1. Entwerfen Sie ein Bewertungsschema mit 5-7 qualitativen Kategorien, die für Ihre gewählte Anwendung relevant sind (z.B. Genauigkeit, Kreativität, Nützlichkeit der Antworten, Verständnis komplexer Anweisungen, Kulturelle Sensibilität).
2. Erstellen Sie für jede Kategorie:
   - Eine klare Definition, was in dieser Kategorie bewertet wird
   - Eine Bewertungsskala (z.B. 1-5 oder 1-10)
   - 2-3 konkrete Testfragen oder -aufgaben, die diese Kategorie prüfen
   - Bewertungskriterien: Was wäre eine ausgezeichnete (5/5) vs. eine unzureichende (1/5) Antwort?
3. Beschreiben Sie den Evaluationsprozess:
   - Wie viele Bewerter sollten eingesetzt werden?
   - Wie würden Sie die Bewertungen zusammenfassen?
   - Welche Maßnahmen würden Sie ergreifen, um Bewertungsverzerrungen zu vermeiden?
4. Erläutern Sie, wie Sie die Ergebnisse dieser qualitativen Bewertung mit quantitativen Metriken (wie MMLU) kombinieren würden, um ein Gesamtbild der Modellleistung zu erhalten.

Abgabeformat:
Ein 2-3 seitiges Konzeptpapier mit Ihrem Evaluationsschema, den Testfragen und dem geplanten Prozess.