

ARBEITSAUFTRAG

## Unit 3: Challenge

Untersuchen Sie, wie zuverlässig GPT-3.5 Turbo eine überprüfbare Aufgabe löst. Messen Sie Fehler und Schwankungen bei wiederholten Aufrufen. Ändern Sie dann einen begründeten Faktor und prüfen Sie die Wirkung.

AUF EINEN  
BLICK

DER  
GESAMTPROZESS

IHRE AUFGABE

VORGEHEN

ARTEFAKTE

INTERPRETATION

EVIDENZ

### Auf einen Blick

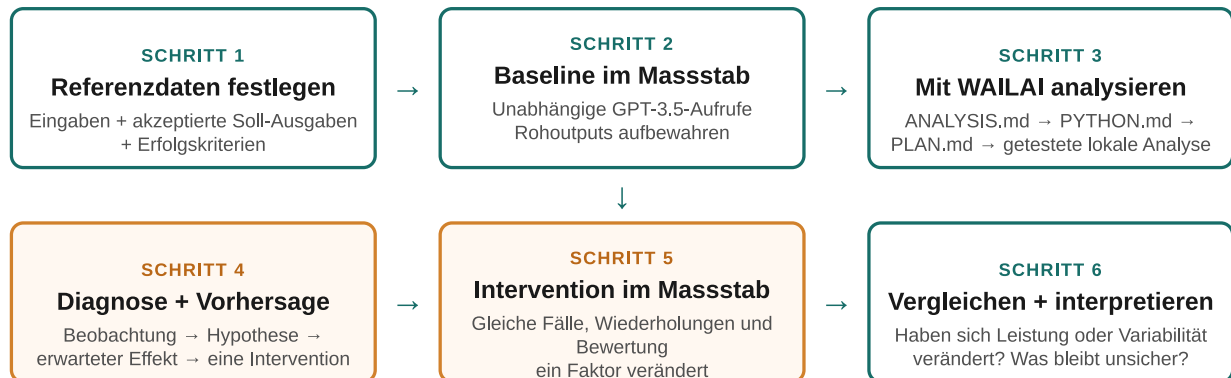
| Frage                | Ihr Auftrag                                                                                                                                          |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------|
| Was erstellen Sie?   | Ein automatisiertes Experiment mit Referenzantworten, gespeicherten Modellausgaben und einer getesteten lokalen WAILAI-Analyse.                      |
| Was vergleichen Sie? | <b>A:</b> einen festen Baseline-Prompt mit festen Einstellungen. <b>B:</b> genau eine daraus begründete Änderung. Jeder Modellaufruf ist unabhängig. |
| Was bleibt gleich?   | Datensatz, Referenzantworten, GPT-3.5 Turbo, Wiederholungen, Bewertungsverfahren und Analysemethode.                                                 |
| Was reichen Sie ein? | Den Projektordner mit Runner, Rohoutputs, Analyse, Ergebnissen und Experimentdossier.pdf.                                                            |

### Der Gesamtprozess

**Standardumfang:** ungefähr 20–50 Fälle, jeder Fall zehnmal pro Bedingung. Beispielsweise ergeben 30 Fälle × 10 Wiederholungen 300 Ausgaben pro Bedingung. Ziel ist ein kontrolliertes, reproduzierbares Experiment. Eine Intervention ohne Verbesserung ist ein gültiges Ergebnis. Bewertet werden das Experiment und seine Belege.

#### Challenge 3 - Referenz, Experiment, Intervention, Evidenz

Konstant: Datensatz, Referenzantworten, GPT-3.5 Turbo, Bewertungsmethode und Wiederholungen. Variiert: eine Intervention.



**Evidenzkette:** Referenzeingabe → GPT-3.5-Aufruf → Rohoutput → Bewertung → aggregiertes Ergebnis

Die sechs Phasen der Abbildung fassen die acht Arbeitsschritte unten zusammen:  
Referenzdaten = Schritt 1; Baseline = Schritte 2–4; WAILAI-Analyse = Schritt 5;  
Diagnose = Schritt 6; Intervention = Schritt 7; Vergleich = Schritt 8.

## Ihre Aufgabe

Wählen Sie eine reale Aufgabe, die Sie automatisiert und systematisch prüfen können. Sie braucht bekannte oder unabhängig geprüfte Soll-Ausgaben, wiederholte unabhängige Modellaufrufe und eine sinnvolle Veränderungsmöglichkeit mit Bezug zu Unit 3.

| Beispiel                             | Aufgabe                                                                                     | Mögliche Veränderung                                              | Mögliche Messung                                                     |
|--------------------------------------|---------------------------------------------------------------------------------------------|-------------------------------------------------------------------|----------------------------------------------------------------------|
| Daten extrahieren                    | Synthetische oder nicht sensible klinische Notizen mit bekannten Angaben in JSON überführen | Genaueres Schema; null bei fehlenden Angaben                      | Feldgenauigkeit, erfundene Angaben, gültiges JSON                    |
| Texte klassifizieren                 | Gelabelte Tickets, Rezensionen oder kurze Dokumente Kategorien zuordnen                     | Temperatur senken oder Kategoriedefinitionen anders strukturieren | Genauigkeit und Konsistenz über Wiederholungen                       |
| Adressen oder Zeichenfolgen umformen | Bekannte Standarddarstellung erzeugen und geschützte Zeichenfolgen erhalten                 | Geschützte Zeichenfolgen markieren oder Eingabe anders darstellen | Exakte Übereinstimmung; Fehler bei Namen, Zahlen und Identifikatoren |

Die Beispiele zeigen mögliche Aufgaben und Messgrößen. Wählen Sie, was zu Ihrer Frage passt.

## Vorgehen

### 1. Referenzdatensatz festlegen

Definieren Sie für jede Eingabe das erwartete Ergebnis **vor den Modellaufrufen**. Geeignet sind gelabelte Daten, fachlich validierte Quellen, manuell erstellte Antworten oder Antworten eines stärkeren Modells, die ein Mensch geprüft und akzeptiert hat. Antworten eines anderen LLMs bleiben Referenzantworten, solange sie nicht unabhängig verifiziert wurden. Erst dann dürfen Sie diese als «Ground Truth» bezeichnen.

Legen Sie vorab fest, welche Ausgabe korrekt, inkorrekt, ungültig oder unbrauchbar ist. Bevorzugen Sie einen exakten Vergleich, wenn möglich. Falls Sie den Inhalt nicht mechanisch prüfen können, legen Sie eine feste Bewertungsrubrik fest.

**Optional:** Ein separates Modell kann freie Texte bewerten. Halten Sie dessen Modell, Prompt, Rubrik und Einstellungen konstant. Prüfen Sie eine Stichprobe seiner Urteile selbst und speichern Sie die Urteile für die spätere Analyse.

Behandeln Sie Eingaben und Referenzdaten als schreibgeschützt.

### 2. Experiment festlegen

Erstellen Sie `.agents/EXPERIMENT.md` mit:

- Aufgabe und Herkunft der Referenzantworten,
- GPT-3.5-Turbo-Version, Baseline-Prompt und Generierungseinstellungen,
- erwartetem Ausgabeformat und Wiederholungen pro Fall,

- Bewertungskriterien und Messgrößen,
- den benötigten Dateien für Rohoutputs und Ergebnisse.

Prüfen Sie, ob diese Angaben Ihre Frage beantworten können. Geben Sie den Plan erst dann für die Ausführung frei. Die Kernfrage lautet: Wie zuverlässig löst das Modell die Aufgabe, und verändert die geplante Intervention Leistung oder Stabilität?

### 3. Experiment-Runner erstellen

Erstellen Sie Code, der die Eingaben liest und für jede Wiederholung einen unabhängigen Modellaufruf ausführt. Das Programm speichert Bedingung, Einstellungen und unveränderte Modellausgabe. Jede Ausgabe muss ihrer Eingabe und Wiederholung zugeordnet bleiben.

Überschreiben Sie keine Referenzdaten. Reparieren Sie Modellausgaben nicht von Hand.

### 4. Baseline ausführen

Führen Sie A vollständig mit dem festgelegten Prompt und den vereinbarten Einstellungen aus. Bewahren Sie die experimentellen Ausgaben auf. Wiederholen Sie fehlerhafte Ergebnisse nicht selektiv, um sie zu ersetzen, und korrigieren Sie sie nicht von Hand.

Die wiederholten Aufrufe zeigen sowohl Korrektheit als auch Schwankungen bei derselben Eingabe.

### 5. Baseline mit WAILAI analysieren

Erstellen Sie mit dem WAILAI-Leitfaden eine reproduzierbare lokale Python-Analyse:

| Datei oder Ordner                | Aufgabe                                                                         |
|----------------------------------|---------------------------------------------------------------------------------|
| <code>.agents/ANALYSIS.md</code> | Was wird gemessen und verglichen? Welche Tabelle und Abbildung werden benötigt? |
| <code>.agents/PYTHON.md</code>   | Was muss die Analysesoftware berechnen und prüfen?                              |
| <code>.agents/PLAN.md</code>     | Welche Meilensteine und Abnahmetests gelten?                                    |
| <code>analysis/</code>           | Ausführbarer Code, Abhängigkeiten und Tests                                     |
| <code>results/</code>            | Geprüfte CSV-Tabelle und mindestens eine aussagekräftige Abbildung              |

Berichten Sie Gesamtleistung, Schwankungen über wiederholte Aufrufe und relevante Fehlerarten oder Untergruppen. Je nach Aufgabe eignen sich etwa Genauigkeit, Precision/Recall, exakte Übereinstimmung, Feldgenauigkeit, Anteil ungültiger Ausgaben oder Konsistenz.

Auch die Bewertung muss reproduzierbar sein: Verwenden Sie dieselbe Bewertungsmethode und bei einem Modellbewerter dessen gespeicherte Urteile und unveränderte Konfiguration.

### 6. Eine Intervention planen

Leiten Sie aus der Baseline eine wichtige Schwäche oder Quelle von Schwankungen ab. Ergänzen Sie **vor Bedingung B** vier Punkte in `EXPERIMENT.md`:

| Punkt        | Leitfrage                                                 |
|--------------|-----------------------------------------------------------|
| Beobachtung  | Welches Muster zeigen die Ergebnisse von A?               |
| Hypothese    | Wie könnte ein Konzept aus Unit 3 dieses Muster erklären? |
| Vorhersage   | Welche Veränderung erwarten Sie bei welchen Ergebnissen?  |
| Intervention | Welchen einzelnen Faktor ändern Sie?                      |

Mögliche Faktoren sind Promptstruktur, Darstellung der Eingabe, Position relevanten Kontexts, Temperatur, Top-p, maximale Ausgabelänge oder eine andere verfügbare Generierungseinstellung. Wählen Sie einen Faktor.

## 7. Intervention ausführen

Führen Sie B für dieselben Fälle und Wiederholungen aus. Ändern Sie nur den vorgesehenen Faktor. Datensatz, Referenzantworten, GPT-3.5 Turbo, Bewertungsverfahren und Analysemethode bleiben gleich. Speichern Sie B getrennt von A.

## 8. Vergleichen und interpretieren

Wenden Sie dieselbe getestete Analyse auf beide Bedingungen an. Prüfen Sie:

- Haben sich Leistung oder Schwankungen verändert?
- Welche Fehlerarten und Fälle sind betroffen? Wo wurde das Ergebnis besser oder schlechter?
- Entspricht das Ergebnis Ihrer Vorhersage?

Ändern Sie die Bewertungskriterien nach Sichtung der Ergebnisse nicht. Stützen Sie Ihre Aussagen auf zusammengefasste Ergebnisse, nicht nur auf ausgewählte Beispiele.

**Beispiel für einen belegten Befund:** «Die Exact-Match-Rate stieg von 68 % bei 300 Baseline-Ausgaben auf 79 % bei 300 Interventions-Ausgaben. Die grösste Veränderung betraf mehrteilige Identifikatoren; die Adressnormalisierung veränderte sich kaum.»

## Artefakte

Reichen Sie den Projektordner mit Experiment-Runner, WAILAI-Analyse, Rohoutputs und `Experimentdossier.pdf` ein. Passen Sie Dateinamen an, wenn Ihre Aufgabe dies sinnvoll macht.

```

project/
├── .agents/
│   ├── EXPERIMENT.md
│   ├── ANALYSIS.md
│   ├── PYTHON.md
│   └── PLAN.md
├── data/
│   ├── inputs.csv
│   └── reference.csv
├── experiment/           Code für die Modellaufrufe
├── outputs/
│   ├── baseline/
│   └── intervention/
├── analysis/           Analysecode, Abhängigkeiten und Tests
├── results/
│   ├── results.csv
│   └── results-figure.png
└── Experimentdossier.pdf

```

Reichen Sie Roh- oder sensible Daten nur ein, wenn dies rechtlich, vertraglich und institutionell zulässig ist. Verwenden Sie für Tests nach Möglichkeit synthetische Daten.

Eine andere Person muss die gesamte Kette prüfen können: **Eingabe** → **Modellaufruf** → **Rohoutput** → **Bewertung** → **zusammengefasstes Ergebnis**. Dazu gehören die Referenzantworten, Prompt und Einstellungen, alle wiederholten Ausgaben mit ihren Bewertungen, die Intervention sowie Analysecode, Tests, Ergebnistabelle und Abbildung.

## Interpretation

Das Dossier beantwortet vier Fragen:

1. **Was ist passiert?** Beschreiben Sie Baseline, Intervention und wichtigste Zahlen. Verweisen Sie auf die stärksten Belege.
2. **Warum könnte es passiert sein?** Beziehen Sie die Beobachtung vorsichtig auf Tokenisierung, Kontext, Next-Token-Generierung oder Sampling. Das Experiment zeigt interne Mechanismen nicht direkt.
3. **Was bleibt unsicher?** Benennen Sie Grenzen von Datensatz, Bewerter, Modell, Wiederholungen und Versuchsdesign. Trennen Sie Beobachtungen von weitergehenden Aussagen.
4. **Was übernehmen Sie künftig?** Formulieren Sie eine konkrete Regel für Ihre LLM-Arbeit, die sich aus den Ergebnissen ableitet.

## Evidenz und Bestehen

Die Challenge ist bestanden, wenn:

- Aufgabe und Referenzantworten dokumentiert sind,
- das Experiment automatisiert mit unabhängigen GPT-3.5-Turbo-Aufrufen durchgeführt wurde,
- Prompt, Modellbedingung, Einstellungen und Wiederholungen vor der Ausführung feststanden,
- die unveränderten Rohoutputs ohne selektive Wiederholungen vorliegen,

- die WAILAI-Analyse spezifiziert, getestet, lokal ausführbar und reproduzierbar ist,
- die Intervention aus A begründet und ihre Vorhersage vor B dokumentiert wurde,
- nur der geplante Faktor verändert und dieselbe Bewertungsmethode verwendet wurde,
- die Schlussfolgerungen durch überprüfbare zusammengefasste Ergebnisse gestützt sind,
- das Dossier Beobachtung, Erklärung, Unsicherheit und Transfer unterscheidet.

Eine Verbesserung ist keine Bestehensbedingung.