KI in Tax, Audit & Advisory

Workshop-Skript zum 12. und 13.05.2026, TH Köln

Prof. Dr. Roman Bartnik

12.05.2026

Inhaltsverzeichnis

1. Willkommen

Skript zu den Workshops am 12. und 13.05.2026

2. Was Sie in zwei Nachmittagen mitnehmen

Nach zwei Workshop-Nachmittagen verfügen Sie über ein gemeinsames Vokabular für den KI-Einsatz in Tax, Audit und Advisory — vom Modellvergleich über das Prompt-Design bis zur Frage, wo *Process Automation* und *Cognitive Automation* sinnvoll zusammenfinden.

2.0.0.1. Am Ende der zwei Workshops können Sie …

1. das **4D-Framework** (Delegate, Describe, Discern, Diligence) erklären und an einem Tax-/Audit-Beispiel anwenden,
2. **Process Automation** und **Cognitive Automation** in Tax-/Audit-Szenarien sauber unterscheiden — auch in Hybrid-Fällen,
3. zwei KI-Modelle und zwei Modell-plus-Tool-Konfigurationen **systematisch vergleichen** und Modell-Effekte von Harness-Effekten trennen,
4. einen Prompt nach **RTF** und **CREATE** strukturieren und einen Tutor-Bot für das eigene Lerngebiet aufsetzen,
5. KI-**Outputs** mit einer kleinen **Test-Suite** systematisch prüfen (Red-Green-TDD nach Willison),
6. eine begründete **persönliche AI Policy** für die spätere Berufspraxis formulieren.

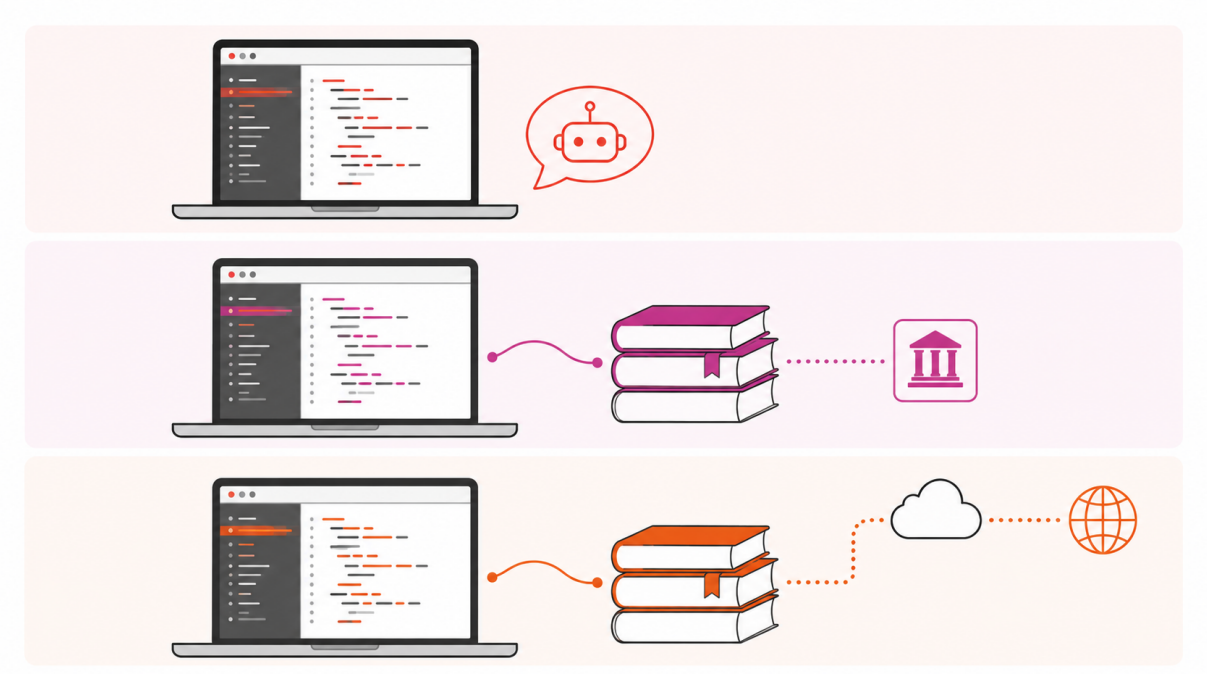


Abbildung 2.1.: Workshop-Rhythmus über zwei Nachmittage — Inputs, eingebettete Übungen, Diskussion.

2.1. Tagesablauf

Termine

- **Workshop 1 — Dienstag, 12.05.2026, 16:00–19:00 Uhr**
- **Workshop 2 — Mittwoch, 13.05.2026, 16:00–19:00 Uhr**
- **Ort** — Campus Südstadt, Claudiusstraße 1 (Raumnummer wird per Mail bekanntgegeben).

Zeitfenster	Workshop 1 (Di) — AI Fluency	Workshop 2 (Mi) — Automatisierung & 4D-Use-Case
16:00–17:30	Inputs und Übungen 1 (Diagnose-Quiz) und 2 (Karriereentwicklung im Modellvergleich)	Recap, BPMN-Grundlagen, UiPath-Demo, BPMN-Übung
17:30–17:40	Pause	Pause
17:40–19:00	Inputs und Übungen 3 (Prompt-Umbau und Tutor-Bot) und 4 (Tutor-Bot systematisch prüfen)	UiPath-Bot-Übung, 4D-Use-Case integriert, Discern-Vertiefung
nach W2	Diligence-Anriss als Brücke zu Tag 2	Hausaufgaben: Dilemma der Mitte (Positionspapier), Personal AI Policy

Tutor öffnen

Tutor — KI in Tax, Audit & Advisory →
Der Tutor begleitet Sie durch alle Übungen. Er kennt das 4D-Framework, die Übungstexte und die Wissensbasis dieses Skripts.

Datenschutz und Vertraulichkeit

In keinem Fall reale Mandantendaten in KI-Tools eingeben. Free-Tier-Dienste sind für Mandantenarbeit grundsätzlich nicht geeignet. Hintergrund: WPO § 43, StBerG § 57, DSGVO. Die berufsrechtliche Vertiefung folgt in Workshop 2.

2.2. Transparenz zur KI-Nutzung

Bei der Erstellung dieses Skriptes wurde **Claude 4.7 / Cowork** zur Aufwertung der Inhalte (speziell für Beispiele und Interaktionen), als Reviewer (speziell Sprach- und Konsistenzchecks) und zur Gestaltung des Layouts genutzt. Die verwendeten Bilder wurden mit **Gemini 3.1** generiert. Alle Inhalte wurden von Roman Bartnik geprüft, überarbeitet und verantwortet.

2.3. Lizenz

Inhalte unter CC BY-NC-SA 4.0 (siehe `LICENSE`). Verwendete Quellen werden inline zitiert und am Ende jedes Kapitels aufgelistet.

Teil I.

Teil 1 — Workshop 1 (Di, 12.05.)

3. Workshop 1 — AI Fluency Framework

Dienstag, 12.05.2026 · 16:00–19:00 · Campus Südstadt

3.1. Lernziele

3.1.0.1. Was Sie nach diesem Workshop können

- *Process Automation* und *Cognitive Automation* an konkreten Tätigkeiten aus Tax, Audit und Advisory unterscheiden — und Grenzfälle als Hybrid benennen,
- das **4D-Framework** (Delegate, Describe, Discern, Diligence) in eigenen Worten erklären und die beiden Loops im Kopf nachzeichnen,
- die drei Konfigurationen **LLM allein**, **erweitertes LLM** und **Agent** unterscheiden und für eine eigene Frage die passende auswählen,
- einen einfachen Prompt nach **RTF** und **CREATE** strukturieren und an einer Aufgabe aus dem eigenen Lerngebiet anwenden,
- einen Tutor-Bot für das eigene Fachgebiet bauen und mit einer kleinen **Test-Suite** systematisch prüfen.

3.2. Inhalte des Workshops

Drei Stunden, eng verzahnt aus fünf kurzen Inputs (zusammen 53 Minuten), vier Übungen mit je 15 Minuten Bearbeitung plus Think-Pair-Share und Plenum-Auflösung (zusammen 88 Minuten), einer Pause (10 Minuten) und Begrüßung/Abschluss (9 Minuten). Alle Übungen sind im Block eingebettet, in dem ihr inhaltlicher Anker liegt — nicht in einen separaten Übungsteil verschoben.

Block 0 — Begrüßung und Kalibrierung · 5 Min

Begrüßung, Lernziele, Hinweis auf den Tutor, kurze Verständigung über Vorerfahrungen mit generativer KI.

Block 1 — Process vs. Cognitive Automation · 10 Min

Process Automation — regelbasiertes Abarbeiten strukturierter Workflows ohne Verstehen. **Cognitive Automation** — KI-gestütztes Schließen, Klassifizieren, Zusammenfassen unstrukturierter Inhalte. Das *Service-Automation-Continuum* nach M. Lacity & Willcocks (2021) und Willcocks & Lacity (2024) ordnet beide Kategorien plus *Intelligent Automation* als orchestrierende Schicht in eine kohärente Sourcing-Strategie ein.

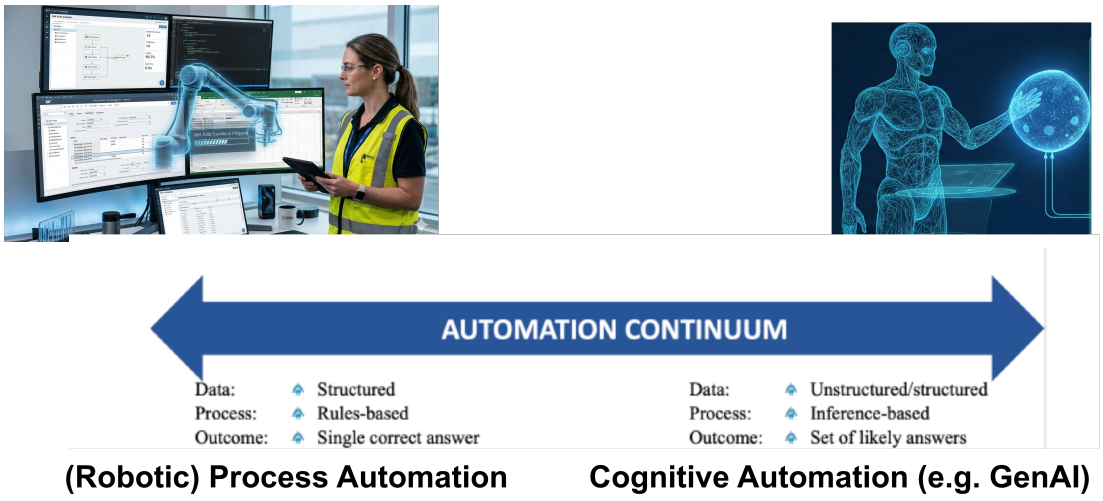


Abbildung 3.1.: Service-Automation-Continuum von Process Automation über Cognitive Automation bis Intelligent Automation. Adaptiert nach Lacity & Willcocks (2021).

Drei Verankerungs-Beispiele aus Tax/Audit: Bank-zu-SAP-Abstimmung (Process); semantische Klassifikation eingehender Belege (Cognitive); Ende-zu-Ende-Bot, der klassifiziert, einbucht und Differenzen meldet (Intelligent Automation).

Übung 1 — Diagnose-Quiz Process vs. Cognitive · 18 Min (8 Min Solo + 4 Min TPS + 3 Min Plenum + 3 Min Puffer)

Block 2 — 4D-Framework Kurzdurchlauf · 7 Min

Vier Kernkompetenzen, je drei Subkategorien nach Dakan & Feller (2025). **Delegate** (Problem-, Platform-, Task-Awareness) · **Describe** (Product-, Process-, Performance-Description) · **Discern** (Product-, Process-, Performance-Discernment) · **Diligence** (Creation-, Transparency-, Deployment-Diligence). Zwei Loops: *Delegate–Diligence* (strategisch), *Describe–Discern* (operativ). Heute behandeln wir Delegate, Describe und Discern in dieser Reihenfolge; Diligence wird morgen vertieft.

Block 3 — Delegate: Modell, Harness, agentische Nutzung · 12 Min

Drei Konfigurationen sauber trennen:

- **LLM allein** — ein Textgenerator. Stellen Sie sich einen sehr belesenen Bibliothekar vor, der nur reden kann, nichts nachschlagen und nichts ausführen.
- **Erweitertes LLM** — dasselbe Modell, eingebettet in eine *Harness* mit Tools wie Web-Search, File-Upload, Code-Execution. Der Bibliothekar bekommt eine Werkbank.
- **Agent** — ein erweitertes LLM, das in einer Schleife arbeitet: Ziel, Plan, Werkzeug, Bewertung, neuer Plan. Der Bibliothekar erledigt eigenständig mehrschrittige Aufgaben und meldet sich erst zurück, wenn er fertig ist.

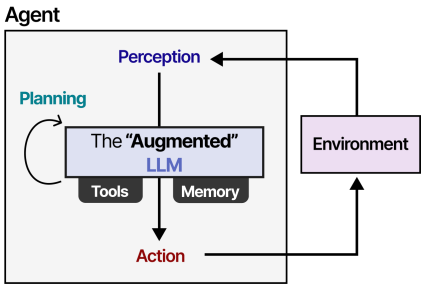


Abbildung 3.2.: LLM-Kern und Harness als Werkzeugring — was ein Modell tatsächlich kann, hängt mindestens so stark von der Harness ab wie vom Modell selbst.

Die didaktische Pointe: Wer Aufgaben delegieren will, muss wissen, in welcher der drei Konfigurationen das eigene System läuft, weil Fähigkeiten und Risikoprofil substanziell verschieden sind (Grootendorst, 2025; Mollick, 2024).

Abbildung 3.3.: Die Jagged Frontier — KI-Fähigkeiten sind unregelmäßig verteilt, mit unerwartet starken und unerwartet schwachen Bereichen. Adaptiert nach Dell'Acqua et al. (2023) und Mollick (2024).

Übung 2 — Karriereentwicklung im Modellvergleich · 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum)

Pause · 10 Min

Block 4 — Describe: RTF und CREATE · 12 Min

Leitanalogie: Der Chatbot ist eher ein junger Nachhilfeschüler als ein Taschenrechner — er versteht Anweisungen, braucht Kontext, möchte gezeigt bekommen, was Sie wollen, und wird mit Beispielen besser. Inhaltlicher Anker: das Kapitel *How to speak* aus dem begleitenden Lehrbuch (Bartnik, 2026).

Zwei Schemata:

- **RTF** — *Role, Task, Format*. Sparsam und schnell. Beispiel: „Sie sind Wirtschaftsprüferin. Erläutern Sie das Going-Concern-Prinzip. Antworten Sie in drei kurzen Absätzen für Erstsemester."
- **CREATE** — *Character, Request, Examples, Adjustments, Type of output, Extras*. Reicher und für komplexe Aufgaben besser geeignet. Beide Schemata mit identischer Beispielaufgabe vorführen, damit der Unterschied direkt sichtbar wird.

Übung 3 — Prompt-Umbau und Tutor-Bot · 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum)

Block 5 — Discern: Outputs systematisch prüfen · 12 Min

Zwei Ideen, eine Übersetzung auf Tax/Audit:

- **RAGAS-Idee** — eine Bewertung von Antworten aus Retrieval-Augmented-Systemen entlang weniger Kerngrößen: *Faithfulness* (passt die Antwort zu den abgerufenen Quellen?), *Answer Relevance* (beantwortet sie die Frage?), *Context Precision* und *Context Recall* (wurden die richtigen Quellen abgerufen?) (Es et al., 2024). Im Kern: ein Test-Set aus Fragen mit erwarteten Antworten, das Sie immer wieder durchlaufen lassen.
- **Red-Green-TDD nach Willison** — Testfälle zuerst. Sie schreiben fünf bis zehn Beispielfragen mit gewünschten Antworten, lassen den Agenten durchlaufen, markieren *rot* (fehlerhaft), *gelb* (teilweise) oder *grün* (korrekt). Dann iterieren Sie Prompt, Modell und Harness, bis möglichst viele Tests grün werden (Willison, 2025). Die Methode kommt aus dem Test-Driven Development der Software-Entwicklung — Kent Becks „Red first" auf Prompt-Ebene übertragen.

Übersetzung auf Tax/Audit: Eine kleine Sammlung typischer Berufsfragen (Anwendung des Reverse-Charge-Verfahrens, Going-Concern-Prüfung, Behandlung von Rückstellungen) wird zur Test-Suite. Genau das bauen Sie in der nächsten Übung — zum Tutor-Bot aus Übung 3.

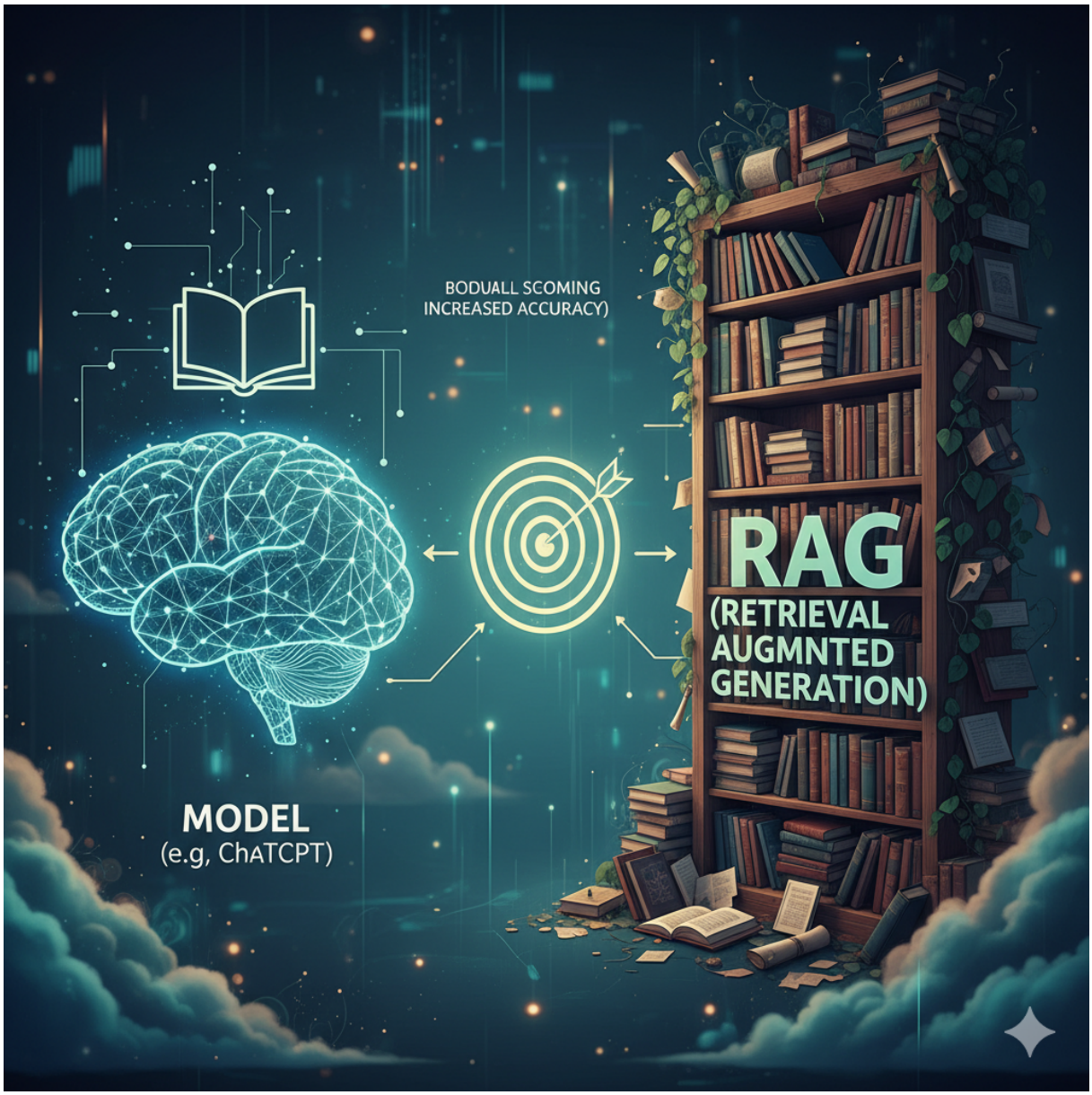


Abbildung 3.4.: RAG als Bücherregal-Metapher — statt aus dem Gedächtnis zu antworten, schlägt das Modell zuerst in einer externen Quelle nach.

Übung 4 — Tutor-Bot systematisch prüfen · 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum)

Block 6 — Abschluss und Brücke zu Tag 2 · 9 Min

Kurzer Rückblick auf die vier Übungen: Welche der drei Konfigurationen (LLM allein, erweitertes LLM, Agent) ist Ihnen heute am vertrautesten geworden? Welche Test-Frage aus Übung 4 würde morgen im echten Berufsalltag bestehen? Brücke zu Workshop 2 (Mittwoch, 13.05.): **Diligence** als vierte Kompetenz — Verantwortung, Transparenz, Verifikation. Konkret morgen: berufsrechtliche Pflichten (WPO § 43, StBerG § 57, DSGVO), Personal AI Policy.

3.3. Diskussionsfragen

- An welcher Stelle in Ihrem Studium würde ein Tutor-Bot Sie heute schon ersetzen — und an welcher nicht?
- Item 9 des Diagnose-Quiz (OCR plus ERP-Einspielung) ist ein Hybrid aus *Cognitive* und *Process*. Wo verläuft die Grenze in einem realen Workflow Ihrer Wahl?
- Welche Konfiguration aus *LLM allein* · *erweitertes LLM* · *Agent* passt zu welcher Aufgabenklasse Ihres Studienalltags? Wo lohnt sich der Aufpreis für ein stärkeres Modell oder eine reichere Harness konkret?
- Wann ist eine Test-Suite mit fünf Fragen aussagekräftiger als eine spontane Qualitätsprüfung? Wann nicht?

3.4. Aufgaben

i Vier Übungen im Workshop, eingebettet in die Inputs

Übung 1 — Diagnose-Quiz Übung 2 — Karriereentwicklung Übung 3 — Prompt-Umbau und Tutor-Bot
Jede Übung dauert maximal 15 Minuten und schließt mit Think-Pair-Share und kurzer Plenum-Auflösung. Innerhalb jeder Übung gibt es eine **Erweiterungsfrage für schnelle Studierende** — wer früh fertig ist, vertieft, statt zu warten. Die Discern-Übung *Tutor-Bot systematisch prüfen* eröffnet Workshop 2 als Übung 2 — Tutor-Bot Test-Suite; davor steht als Einstieg in den zweiten Tag der zweistufige Quellenbindungs-Prompt an der eigenen Prüfungsordnung.

3.5. Hausaufgabe zum Folgetag

Für Workshop 2 brauchen Sie zwei kostenfreie Accounts. Bitte vor der Sitzung am Mittwoch einrichten — beides dauert jeweils etwa fünf Minuten.

i Zwei Accounts vor Workshop 2

- **Signavio Academic Edition** — Account anlegen unter https://academic.signavio.com/p/explorer. Wird in Workshop 2 für die BPMN-Modellierung des Rechnungseingangsprozesses genutzt. Hochschul-E-Mail-Adresse wird empfohlen, kostenfrei, kein Download nötig.
- **UiPath Cloud** — Account anlegen unter https://cloud.uipath.com/. Wird für das Lesen und Anpassen des Excel-zu-PDF-Bots genutzt. Free-Tier reicht; *Studio Web* läuft direkt im Browser, kein lokales Studio nötig.

Bitte zu Workshop 2 mitbringen: beide Logins (E-Mail und Passwort gespeichert oder im Browser eingeloggt). Wer keinen Account anlegen kann, schaut beim Sitznachbarn mit — verpasst aber das Hands-on.

3.6. Hintergrund / Werkzeuge / Ressourcen

- **Originalmaterial 4D-Framework** — Cheat Sheet und Practical Overview von Dakan & Feller (2025).
- **Begleitlehrbuch** — Kapitel 2.4 *How to speak* und Kapitel 7 *Tutor-Prompt-Sammlung*.
- **O*NET-Berufsbild** — Accountants and Auditors (13-2011.00) als Faktenbasis für Übung 2.
- **Vergleichsplattform** — arena.ai mit Side-by-Side-Modus für Übung 2.

3.7. Weiterführend

- Mollick (2024) — *Co-intelligence: Living and working with AI* (Begleitlektüre zum Studium).
- Dell'Acqua et al. (2023) — *Navigating the jagged technological frontier* (HBS Working Paper, empirisch zur Produktivitätswirkung).
- Grootendorst (2025) — *A visual guide to LLM agents* (visuelle Erklärung des Harness-Konzepts und der drei Konfigurationen).
- Willison (2025) — *Red-green test-driven development for agentic engineering* (didaktisch klare Einführung in TDD für KI-Systeme).
- Es et al. (2024) — *RAGAS: Automated evaluation of retrieval augmented generation* (Originalpaper zur Bewertung von RAG-Systemen).

3.8. Tutor

Tutor öffnen

Tutor — KI in Tax, Audit & Advisory →
Vorschlag-Prompt für diesen Workshop:

> Ich bereite Workshop 1 (AI Fluency Framework) im Modul *KI in Tax, Audit & Advisory* vor. Bitte erklären Sie mir die Unterscheidung *Process Automation* versus *Cognitive Automation* an einem konkreten Beispiel meiner Wahl: [Beispiel einfügen]. Fragen Sie mich nach meinem Vorwissen, geben Sie eine kurze Lay-Erklärung vor jedem Fachbegriff und stellen Sie am Ende drei Diagnose-Fragen, mit denen ich mein Verständnis selbst prüfen kann.

4. Übungen Workshop 1

Drei in den Workshop eingebettete Übungen, je 15 Minuten plus Auswertung

4.0.0.1. Was Sie nach diesen drei Übungen können

- Tätigkeiten als *Process*, *Cognitive* oder *Nicht anwendbar* klassifizieren — auch in Grenzfällen begründen,
- zwei KI-Modelle und zwei Modell-plus-Tool-Konfigurationen an einer eigenen Frage **systematisch vergleichen** und Modell- von Harness-Effekten trennen,
- einen Prompt in die Schemata **RTF** und **CREATE** umbauen und einen **Tutor-Bot** für das eigene Lerngebiet aufsetzen.

Jede Übung dauert maximal 15 Minuten Bearbeitung plus kurzes Think-Pair-Share. Innerhalb der 15 Minuten steht eine **Erweiterung für schnelle Studierende** — wer früh fertig ist, vertieft. Die ursprünglich für Workshop 1 vorgesehene Discern-Übung *Tutor-Bot systematisch prüfen* eröffnet jetzt Workshop 2 als Übung 2 — Tutor-Bot Test-Suite; davor steht als Einstieg die Übung am zweistufigen Quellenbindungs-Prompt an der eigenen Prüfungsordnung.

4.1. Übung 1 — Diagnose-Quiz Process vs. Cognitive

Modus: in-class · Einzelarbeit · **Dauer:** 18 Min (8 Min Solo + 4 Min TPS + 3 Min Plenum + 3 Min Puffer) · **Anker:** Block 1.

Klassifizieren Sie die zehn Items des Diagnose-Quiz spontan in acht Minuten. Spontane Einschätzung, keine Recherche. Quiz im Browser starten: Diagnose-Quiz. Notieren Sie für strittige Items zwei bis drei Stichworte. Anschließend Think-Pair-Share mit dem Sitznachbarn, danach Plenum-Auflösung mit Balken-Diagramm. Der strittige Hybrid-Fall (Item 9 — OCR plus ERP-Einspielung) ist der Diskussions-Anker.

Erweiterung für schnelle Studierende. Schlagen Sie für ein Item, das Sie als Hybrid einordnen würden, eine klare Trennlinie vor: An welcher Stelle des Workflows endet *Process Automation* und beginnt *Cognitive Automation*? Welches technische Element markiert den Übergang (OCR-Modul, Klassifikator, Regel-Engine)?

Think-Pair-Share (4 Min). Welches Item war für Sie eindeutig, welches strittig? Wo verläuft die Grenze, wenn beide Kategorien in einem Workflow zusammenwirken?

4.2. Übung 2 — Karriereentwicklung im Modellvergleich

Modus: in-class · Einzelarbeit · **Dauer:** 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum) · **4D-Bezug:** Delegate.

Stellen Sie dieselbe Frage an ein schwächeres und ein stärkeres Modell und vergleichen Sie anschließend zwei Modelle mit Suchwerkzeug im Side-by-Side. Schwächeres Modell: Meta Llama 3.1 8B Instruct in der GWDG Academic Cloud. Stärkeres Modell: ein Free-Account bei Claude, ChatGPT, Gemini oder Perplexity — Ihre Wahl. Side-by-Side-Vergleich auf arena.ai. Faktenbasis: O*NET-Profil 13-2011.00 Accountants and Auditors.

Schritt a — Karriereberatung im Direktvergleich (5 Min). Dieselbe Frage an Llama 3.1 8B und an Ihr starkes Modell stellen:

> Berate mich: Ich studiere Tax, Audit und Advisory. Wie sollte ich mich auf die Auswirkungen generativer KI auf den Arbeitsmarkt für Accountants und Auditors vorbereiten? Bezug: O*NET-Profil 13-2011.00.

Drei Stichworte pro Modell zur Qualität notieren (Tiefe, Konkretheit, Belege).

Schritt b — KPMG-Überblick mit Suchwerkzeugen (7 Min). Auf arena.ai *Side by Side* öffnen. Linke Seite: Llama 3.1 8B. Rechte Seite: zwei Search-fähige Modelle über die Globus-Schaltfläche auswählen — `grok-4.20-multi-agent` und `gpt-5.2-search`. Dieselbe Frage an alle drei Konfigurationen:

> Geben Sie mir einen Überblick zu aktuellen Entwicklungen im Geschäft von KPMG Deutschland, mit besonderem Fokus auf die Nutzung generativer KI und die wirtschaftliche Lage.

In drei Stichworten notieren: Welche Output-Unterschiede gehen auf das **Modell** zurück, welche auf die **Harness** (Web-Search)? Wo halluziniert Llama erkennbar?

Erweiterung für schnelle Studierende. Einen Deep-Research-Lauf vorbereiten und starten. Tools nach Wahl (GPT Plus mit *Deep Research*, Gemini Advanced, Perplexity Pro im Trial). Zuerst lassen Sie sich vom starken Modell den Deep-Research-Prompt erzeugen — Meta-Prompt:

> Erstellen Sie mir einen detaillierten Prompt nach dem RTF-Schema für eine Deep-Research-Analyse von Praxisbeispielen aus hochwertiger grauer Literatur (vor allem Think Tanks, Reports der großen Unternehmensberatungen und Big Four, Case Studies von Anbietern wie Salesforce oder IBM) zur Aufgabenstellung: „Process Automation und Cognitive Automation im Berufsfeld Accountants and Auditors — Erfolgsfaktoren, Barrieren, Best Practices." Fragen Sie mich anfangs nach weiteren Details, wenn nötig. Bauen Sie Prüfschritte gegen Halluzinationen ein.

Dann den erzeugten Prompt in den Deep-Research-Modus geben und den Bericht abwarten.

Think-Pair-Share (4 Min). Welche Unterschiede zwischen schwachem und starkem Modell ließen sich klar auf die Modellqualität zurückführen, welche auf die Harness? Wo lohnt sich der Aufpreis für ein stärkeres Modell konkret in Ihrer Studien- und Berufspraxis?

4.3. Übung 3 — Prompt-Umbau und Tutor-Bot

Modus: in-class · Einzelarbeit · **Dauer:** 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum) · **4D-Bezug:** Describe.

Bauen Sie einen Beispiel-Prompt in die Schemata RTF und CREATE um und setzen Sie anschließend einen Tutor-Bot für das eigene Lerngebiet auf. Der Tutor-Bot wird in Übung 4 mit einer Test-Suite geprüft — geben Sie ihm einen Namen, den Sie wiederfinden. Setup: ein Free-Account bei Claude, ChatGPT, Gemini oder Perplexity mit *Custom Instructions*, *Projects* oder *Gems*. Optional für die Erweiterung: NotebookLM (notebooklm.google.com) und ein bis drei fachliche Quellen zum Hochladen.

Schritt a — Prompt in RTF und CREATE umbauen (7 Min). Ausgangs-Prompt:

> Du generierst klare, genaue Beispiele für Konzepte für Studierende. Ich möchte, dass du mir zwei Fragen stellst: Welches Konzept soll ich erklären, und wer ist die Zielgruppe für die Erklärung. Schlage dann das Konzept und Beispiele für das Konzept nach. Liefere eine klare, mehrteilige Erklärung des Konzepts unter Verwendung spezifischer Beispiele und gib mir fünf Analogien und Metaphern, die ich verwenden kann, um das Konzept auf unterschiedliche Weise zu verstehen.

Strukturieren Sie diesen Prompt einmal nach **RTF** (Role, Task, Format) und einmal nach **CREATE** (Character, Request, Examples, Adjustments, Type of output, Extras). Notieren Sie eine Beobachtung pro Schema.

Schritt b — Tutor-Bot anpassen (8 Min). Vorlage: General-Tutor-Prompt aus der Prompt-Sammlung. Kopieren Sie den Prompt in *Custom Instructions*, *Project Instructions* oder *Gem*. Anpassung auf Ihr Lerngebiet — Steuerberater-Examen, Wirtschaftsprüfer-Examen oder ein Modul des laufenden Semesters. Testen Sie den Tutor mit zwei Beispielfragen.

Erweiterung für schnelle Studierende. RAG-Tutor mit NotebookLM. In NotebookLM ein neues Notebook anlegen, ein bis drei fachliche Quellen hochladen, denselben Tutor-Prompt verwenden. Wie ändert sich die Antwortqualität durch die externe Wissensquelle?

Think-Pair-Share (4 Min). Welches Schema (RTF oder CREATE) passt zu welchem Aufgabentyp? Wo merken Sie, dass Ihr Tutor-Bot an Grenzen stößt, und was würde **RAG** daran ändern?

4.4. Quellen

Dakan & Feller (2025) · M. Lacity & Willcocks (2021) · Willcocks & Lacity (2024) · Grootendorst (2025) · Mollick (2024) · Bartnik (2026) · Es et al. (2024) · Willison (2025) · Dell'Acqua et al. (2023)

Teil II.

Teil 2 — Workshop 2 (Mi, 13.05.)

5. Workshop 2 — Prozessmodellierung, Automatisierung, 4D-Use-Case

Mittwoch, 13.05.2026 · 16:00–19:00 · Campus Südstadt

5.1. Lernziele

5.1.0.1. Was Sie nach diesem Workshop können

- Sie können einfache Qualitätsprüfungen von GenAI-Outputs in Prompts einbauen (**Qualitätsprüfung durch Prompt-Engineering**).
- Sie verstehen die Möglichkeiten und Grenzen von RAG und Techniken der detaillierteren Qualitätsprüfung (**deterministische Prüfung, RAGAS, Goldstandard**).
- Sie verstehen die drei Säulen von **Diligence** (Creation, Transparency, Deployment).
- Sie können einen einfachen Prozess modellieren (Aktivität, Entscheidungspunkt, Swimlane).
- Sie können einen einfachen **Bot in UiPath** lesen und in Grundzügen erklären.

5.2. Vorbereitung

- **Zugang** zur BPMN-Modellierungsumgebung (Signavio Academic Edition oder Alternative wie bpmn.io) prüfen, alternativ Stift und Papier.
- **UiPath-Account** in der Cloud anlegen oder Screencast-Ersatzmaterial bereitstellen — siehe Quickstart.
- Optional: die Wissensbasis zu RAG, RAGAS und Prompt-Patterns überfliegen.

5.3. Inhalte des Workshops

Block 0 — Recap und Brückenfrage · 5 Min

Was ist aus Workshop 1 hängen geblieben? Brücke von **Describe** (Tag 1) zu **Discern** und **Diligence** (Tag 2) — von der präzisen Beschreibung zur systematischen Output-Prüfung und persönlichen Verantwortung.

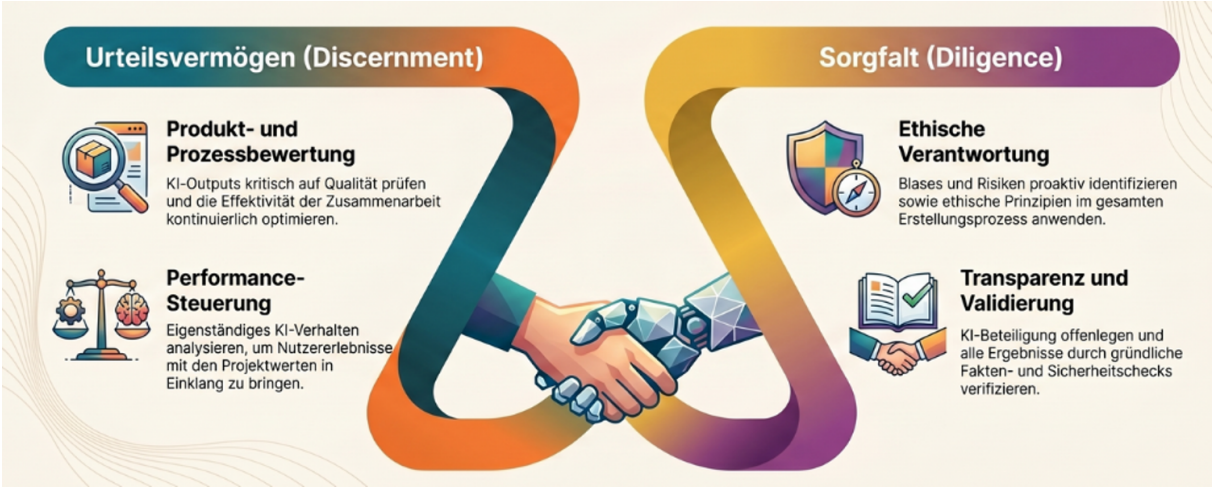


Abbildung 5.1.: Vom Discern zur Diligence — die Output-Prüfung am Tisch wird im Berufsalltag zur dokumentierten Verantwortung.

Block 1 — Qualitätssicherung von GenAI-Outputs in drei Schichten · 15 Min

Qualitätssicherung von GenAI-Outputs folgt dem Prinzip *Defense in Depth* — keine einzelne Maßnahme genügt, die Schichten fangen jeweils andere Fehlerklassen ab (Reason, 2000). Drei Schichten greifen ineinander.

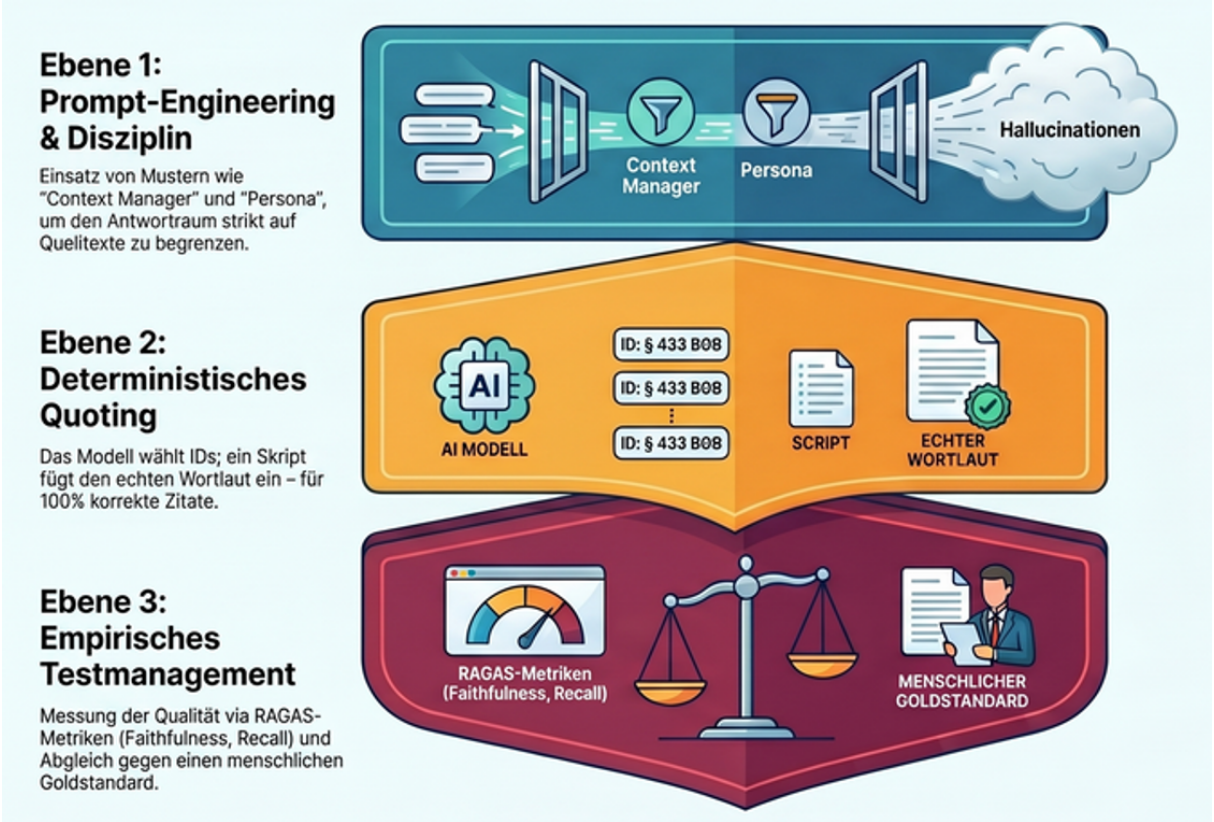


Abbildung 5.2.: Drei Ebenen der RAG-Qualitätsprüfung — Prompt-Engineering, deterministische Prüfung und Goldstandard-Tests greifen wie Schichten einer Defense in Depth ineinander.

Schicht 1 — Prompt-Engineering diszipliniert das Modell innerhalb seiner Antwort. Konkret

nach White et al. (2023) sechs Patterns aus dem Prompt-Pattern-Katalog: **Context Manager** begrenzt den Antwortraum auf den retrieveten Text und verbietet Rückgriff auf Trainingswissen. **Persona** weist dem Modell die Rolle eines wortgetreuen Textanalysten ohne Entscheidungskompetenz zu. **Template** erzwingt eine feste Ausgabestruktur (Sachverhalt → einschlägige Stellen → Auslegungsoptionen → Grenzen → Fact Check → Selbstprüfung). **Fact Check List** zwingt das Modell, am Ende offenzulegen, welche Aussagen Folgerungen sind und gegengeprüft werden müssen. **Reflection** instruiert zur Selbstprüfung vor Abgabe. **Alternative Approaches** verlangt mehrere Auslegungsoptionen statt einer Entscheidung. Diese Schicht senkt Halluzinationen, kontrolliert aber nicht den Wortlaut der zitierten Stellen.

Schicht 2 — Deterministische Prüfung nimmt dem Modell den Stift aus der Hand. *Deterministic Quoting* nach Yeung (2024) trennt die Aufgabe „richtige Stelle auswählen" (Sprachmodell) von „Wortlaut wiedergeben" (deterministisches Skript). Das Modell markiert Zitate mit kanonischen IDs (z. B. `HGB-§267-Abs1-Satz1`); ein nachgeschaltetes Skript schlägt den echten Wortlaut im Index nach und überschreibt den vom Modell gelieferten Text. Existiert die ID nicht im Index, ist sie halluziniert. Yeung berichtet für den so geprüften Bereich *zero false positives*.

Schicht 3 — Testmanagement misst die statistische Qualität des Systems über viele Antworten. Ein einfaches und weitgehend automatisiertes Werkzeug dafür: **RAGAS** (Es et al., 2024) mit vier Kernmetriken — *Faithfulness* (passt die Antwort zu den Quellen?), *Answer Relevance* (beantwortet sie die Frage?), *Context Precision* (sind die relevanten Chunks weit oben?), *Context Recall* (wurden alle relevanten Chunks gefunden?). Wichtige Warnung: die Faithfulness-Bewertung selbst wird von einem Sprachmodell vorgenommen und unterschätzt systematisch die tatsächliche Halluzinationsrate (Magesh et al., 2025). RAGAS taugt für *Grobeinschätzung der Qualität* und *relative Bewertung* (wird mein System schlechter, wenn ich das Modell oder den Prompt ändere?), nicht für absolute Zertifizierung. Einen absoluten Befund liefert nur ein manuell kuratierter **Goldstandard** — eine Sammlung typischer Berufsfragen mit expertengeprüften Soll-Antworten, gegen die das System regelmäßig läuft.

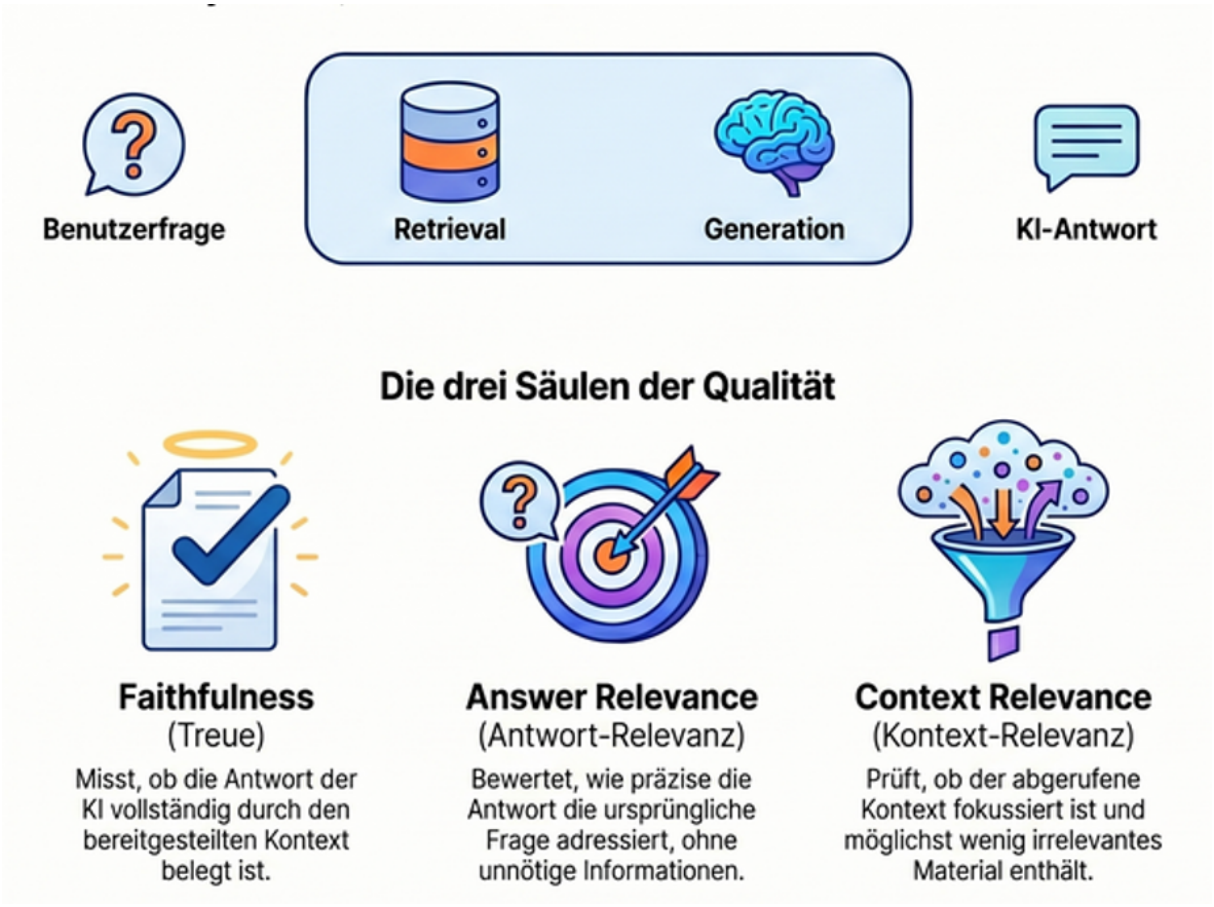


Abbildung 5.3.: RAGAS-Metriken und Goldstandard im Zusammenspiel — automatisierte Grobeinschätzung trifft auf expertengeprüften Referenzkanon.

Vertiefung in der Wissensbasis · Primer Qualitätsprüfung und Primer Goldstandard. Pattern-Details unter Prompt-Pattern-Katalog.

Block 2 — Diligence vertieft · 15 Min

Drei Säulen (Dakan & Feller, 2025):

Creation Diligence — verantwortlicher Umgang während der Erzeugung. Verantwortungsvoller Umgang mit KI-Tools unter Einhaltung ethischer und rechtlicher Best Practices; Bewusstsein für Verzerrungen, Mängel, Auswirkungen auf Interessengruppen; Fähigkeit, Verzerrungen und ethische Risiken in KI-generierten Inhalten zu erkennen und zu mindern. *Ziel:* Gewährleistung eines verantwortungsvollen und sozialbewussten Einsatzes von KI.

Transparency Diligence — Offenlegung gegenüber Stakeholdern. Transparenz und Verantwortlichkeit bei der Verbreitung des Endprodukts; Verständnis für die Erwartungen und Normen des Publikums, der Branche und der Rechtsordnung; Fähigkeit, die Art der KI-Beteiligung klar zu kommunizieren. *Ziel:* Wahrung von Vertrauen und Integrität. EU AI Act Art. 50 setzt seit 2025 explizite Transparenzpflichten für generative Outputs.

Deployment Diligence — Verantwortung für Verifikation und Veröffentlichung. Verantwortung für die Verifikation von KI-Outputs übernehmen, einschließlich gründlicher Faktenprüfung; angemessene Sicherheitsprüfungen vor der Freigabe; Risiken und Auswirkungen veröffentlichter KI-Inhalte verstehen und verantworten. *Ziel:* Sicherstellung von Qualität, Sicherheit und Verlässlichkeit. Im Berufsstand übersetzt: Wer haftet, wenn der KI-Output Schaden anrichtet? Antwort: immer Sie als Berufsträger:in, nie der Anbieter.

Berufsrechtlicher Rahmen für Tax/Audit/Advisory:

- **WPO § 43** — Berufsgrundsätze für Wirtschaftsprüfer: Eigenverantwortlichkeit, Gewissenhaftigkeit, Verschwiegenheit, Unabhängigkeit.
- **StBerG § 57** — gleichgelagerte Pflichten für Steuerberater. Verschwiegenheit ist eine Berufspflicht, kein Vertragsthema; ihre Verletzung ist Straftat nach § 203 StGB.
- **DSGVO Art. 5** — Datenschutz-Grundsätze, insbesondere *Zweckbindung* und *Datenminimierung*. Free-Tier-Anbieter trainieren in der Regel auf Eingaben — damit ist Mandantenbezug ausgeschlossen.

Block 3 — BPMN und Prozessmodellierung · 15 Min

BPMN (Business Process Model and Notation) ist eine bildliche Sprache zur Beschreibung von Geschäftsprozessen — Aktivitäten als abgerundete Rechtecke, Entscheidungen als Rauten, Verantwortlichkeiten als Schwimmbahnen (Object Management Group, 2014). Bezug zu Workshop 1: BPMN ist die *Sprache*, in der wir entscheiden, welche Aktivitäten sich für *Process Automation* (Bot) und welche für *Cognitive Automation* (Sprachmodell) eignen — das setzt das gestrige Diagnose-Quiz operativ um.

Für die heutige Übung beginnen Sie niederschwellig mit **Mermaid** (mermaid.live) als textbasierter Notation und wechseln dann zu **Signavio Academic Edition** für die standardisierte BPMN-2.0-Modellierung mit Swimlanes.

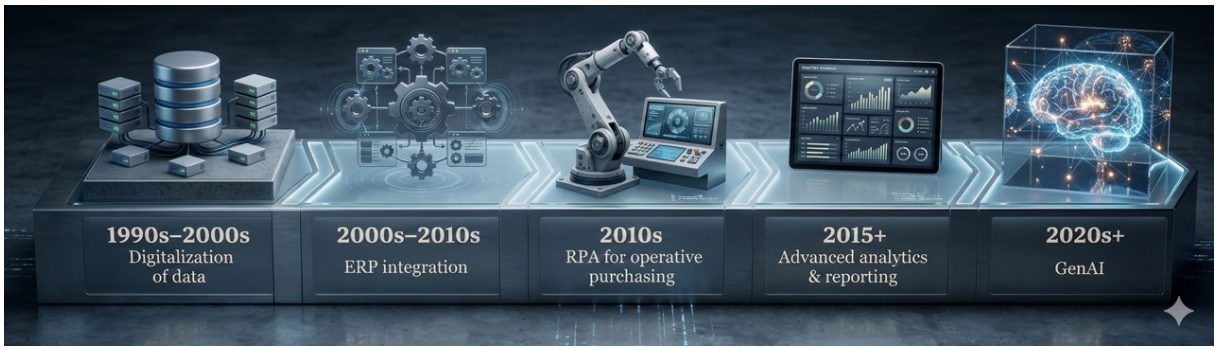


Abbildung 5.4.: Automatisierungs-Generationen — die heutigen Probleme mit GenAI ähneln denen, die ERP, RPA und Advanced Analytics in ihren Anfangsjahren hatten.

Direkt zur Hand (3 Min): Skizzieren Sie auf Papier den Prozess *„Spesenabrechnung freigeben"* mit höchstens fünf Aktivitäten und einem Entscheidungspunkt — Vergleich am Tisch.

Block 4 — Process Automation mit UiPath · 10 Min

RPA (Robotic Process Automation) — Software-Roboter, die genau das tun, was ein Mensch in einer Anwendung tun würde, nur reproduzierbar und ohne Ermüdung. **UiPath** ist eine der führenden RPA-Plattformen; mit **Studio Web** läuft die Entwicklung browserbasiert. Bezug zum gestrigen Diagnose-Quiz: Die Items, die Sie als *Process Automation* eingeordnet haben, sind die Kandidaten für UiPath-Bots.

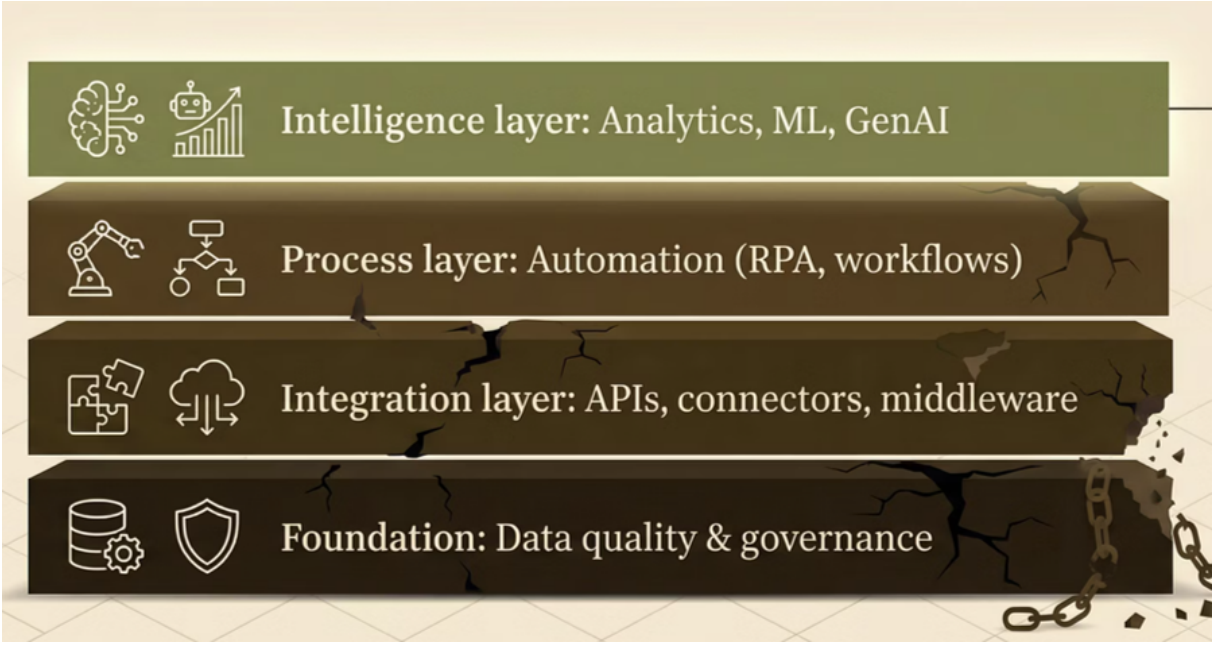


Abbildung 5.5.: GenAI-Tech-Stack mit Datenqualität, Tool-Integration und Process Intelligence als Fundament. Adaptiert nach Willcocks & Lacity (2024).

Direkt zur Hand (2 Min): Lassen Sie sich von Ihrem starken Modell beschreiben, wie ein einfacher Mandanten-Anschreiben-Bot in UiPath aufgebaut wäre — Excel-Adressliste lesen, Word-Vorlage befüllen, Datei speichern. Anschließend in der Übung mit der Studio-Web-Oberfläche abgleichen.

5.4. Diskussionsfragen

- Welche der drei Qualitätssicherungs-Schichten ist in Ihrem Berufsfeld am ehesten praktisch umsetzbar — Prompt-Engineering, Deterministic Quoting oder Goldstandard-Tests?
- An welcher Stelle Ihres BPMN-Modells wäre *Cognitive Automation* sinnvoller als *Process Automation*?
- Welche Diligence-Pflicht aus WPO § 43, StBerG § 57 oder DSGVO Art. 5 wird in Ihrer typischen KI-Nutzung am ehesten verletzt — und an welcher Stelle des Workflows?
- Wie würden Sie *Transparency Diligence* in der Mandantenkommunikation dokumentieren, ohne den Mandanten zu verunsichern?

5.5. Aufgaben

Sieben Übungen plus Vertiefung und Hausaufgaben

In-class — Discern und Diligence zuerst, dann Modellierung und Automatisierung:
Übung 1 — Prüfungsordnung mit zweistufigem Dialog-Prompt (Describe + Discern) Übung 2 — Tutor-Bot systematisch prüfen (Discern) Übung 3 — RAG-Suchübung HGB-Prüfungspflicht (Discern) Übung 4 — Diligence-Risiko-Recherche im Best-of-N (Diligence) Übung 5 — Prozessmodellierung mit Mermaid Übung 6 — BPMN-Modellierung in Signa-

vio Übung 7 — UiPath Mandanten-Anschreiben
Vertiefung (optional in-class oder als Hausaufgabe):
Übung 8 — Integrierter 4D-Use-Case UiPath-Vertiefung Excel-zu-PDF (Hausaufgabe)
Die ersten vier Übungen vertiefen **Discern** (zweistufiger Quellenbindungs-Prompt an der Prüfungsordnung, Test-Suite am Tutor-Bot aus W1, RAG-Suchhilfe im HGB) und **Diligence** (Best-of-N-Risikorecherche). Anschließend gewöhnen Sie sich an Prozessmodellierung — erst niedrigschwellig mit **Mermaid**, dann fachgerecht in **Signavio**. Den Abschluss bildet das Kennenlernen von **UiPath Studio Web** an einem Mandanten-Anschreiben-Bot.

5.6. Hintergrund / Werkzeuge / Ressourcen

- **Wissensbasis** — Prompt-Pattern-Katalog nach White · RAG-Grundlagen · Primer Qualitätsprüfung · Primer Goldstandard · BPMN-Grundlagen · RPA-Grundlagen · Recht und Berufsstand.
- **Originalmaterial** — Cheat Sheet und Practical Overview zu Diligence aus Dakan & Feller (2025).
- **Standardreferenz BPMN** — Object Management Group (2014).
- **Service-Automation-Continuum** — M. Lacity & Willcocks (2021) und Willcocks & Lacity (2024) als Vertiefung zur strategischen Einordnung.

5.7. Weiterführend

- White et al. (2023) — *A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT* (Originalpaper zu den 16 Patterns).
- Yeung (2024) — *Deterministic Quoting* als praktisches Verfahren gegen Zitat-Halluzinationen.
- Es et al. (2024) — *RAGAS: Automated Evaluation of Retrieval Augmented Generation.*
- Magesh et al. (2025) — empirische Studie zu Halluzinationsraten in juristischen RAG-Systemen.
- M. C. Lacity & Willcocks (2016) · Coombs et al. (2020) · Davenport & Ronanki (2018) — drei klassische Texte zur Service-Automatisierung und intelligenter Automation in Wissensarbeit.

5.8. Tutor

Tutor öffnen

Tutor — KI in Tax, Audit & Advisory →
Vorschlag-Prompt für diesen Workshop:

> Ich arbeite in Workshop 2 am Block *Qualitätssicherung von GenAI-Outputs.* Erklären Sie mir am Beispiel eines HGB-RAG-Systems, wie die drei Schichten (Prompt-Engineering nach White, Deterministic Quoting nach Yeung, Goldstandard-Tests mit RAGAS) ineinandergreifen. Welche Schicht würden Sie zuerst aufbauen, wenn ich morgen anfangen müsste — und warum?

6. Übungen Workshop 2

Discern und Diligence zuerst, dann Modellierung und Automatisierung

6.0.0.1. Was Sie nach diesen Übungen können

- die eigene **Prüfungsordnung** mit einem zweistufigen Dialog-Prompt quellengebunden befragen und die Output-Qualität gegen ein Prüfprotokoll abklopfen,
- einen **Tutor-Bot** mit einer kleinen Test-Suite systematisch prüfen — Vertiefung aus Workshop 1,
- eine **RAG-Suche im HGB** so einhegen, dass die KI nur als quellengebundene Suchhilfe agiert,
- aus einer **Diligence-Definition** mehrere Risikofälle im Best-of-N-Schema generieren und einen aussagekräftigen auswählen,
- einen einfachen Prozess in **Mermaid** und anschließend in **Signavio** als BPMN-Diagramm modellieren,
- die **Grundstruktur von UiPath** an einem Mandanten-Anschreiben-Bot kennenlernen.

Die Reihenfolge folgt einer didaktischen Logik: erst drei Discern-Übungen am eigenen Dokument, am Tutor-Bot und im HGB, dann eine Diligence-Übung, dann der Übergang zur technischen Brücke aus Prozessmodellierung und Robotic Process Automation. Die Vertiefungen zu UiPath und zum integrierten 4D-Use-Case sind als Hausaufgabe oder optionale Übung am Ende dieser Sektion aufgeführt.

6.1. Übung 1 — Prüfungsordnung mit zweistufigem Dialog-Prompt befragen

Modus: in-class · Einzelarbeit · **Dauer:** 30 Min · **4D-Bezug:** Describe und Discern — präziser Prompt plus quellengebundene Output-Prüfung am eigenen Dokument.

Sie öffnen Workshop 2 an einem Dokument, das Sie kennen: Ihre eigene Prüfungsordnung. Die Aufgabe — ein quellengebundener Analyse-Assistent liefert Ihnen mehrere Lesarten zu einer konkreten Frage, mit wörtlichen Zitaten und Fundstellen, ohne abschließendes Urteil. Der Witz dabei: der Prompt zerlegt den Dialog in *zwei* Schritte. Erst nimmt das Modell das Dokument nur auf und benennt es, dann erst beantwortet es die Frage. So vermeiden Sie, dass das Modell zu früh interpretiert oder spekuliert. Übersetzt in den Berufsalltag: dasselbe Muster passt auf Mandanten-Mandate, Verträge und Bescheide.

Setup. Ein starkes Modell mit Dokument-Upload (ChatGPT mit Datei-Upload, Claude Projects, NotebookLM oder GWDG Academic Cloud RAG/Arcana). Ihre aktuelle Prüfungsordnung oder Modulordnung als PDF bereitlegen. **Keine personenbezogenen Daten über sich**

oder Dritte in den Chat eingeben — die Übung läuft mit dem Dokument und einer fachlichen Frage.

Schritt 1 — Zweistufigen System-Prompt setzen (3 Min). Kopieren Sie den unten stehenden Prompt als System- oder Initial-Prompt in den Chat. Der Prompt erzwingt den Dialog in zwei Schritten.

System-Prompt zum Kopieren

Du bist ein quellengebundener Analyse-Assistent für Prüfungsordnungen im Hochschulkontext. Deine Aufgabe ist, Studierenden zu helfen, ihre eigene Prüfungsordnung zu verstehen. Du gibst keine endgültige Rechtsauskunft und triffst keine verbindliche Entscheidung. Du arbeitest in zwei Schritten.

Schritt 1 — Dokument aufnehmen. Wenn ein Dokument hochgeladen wurde, tue nur Folgendes: (1) bestätige, dass das Dokument als Grundlage verwendet wird; (2) benenne — soweit erkennbar — Titel, Studiengang/Prüfungsordnung/Rahmenordnung, Datum/Fassung/Version und Struktur (Paragraphen, Abschnitte, Anlagen); (3) weise darauf hin, dass du ausschließlich mit dem hochgeladenen Dokument arbeitest; (4) fordere zu einer konkreten Frage auf. Beantworte in Schritt 1 noch keine inhaltliche Rechtsfrage, interpretiere noch nicht, suche noch keine Regelungen heraus, erfinde keine Angaben.

Schritt 2 — Frage analysieren. Wenn der User anschließend eine konkrete Frage stellt, analysiere sie ausschließlich anhand des hochgeladenen Dokuments. Du darfst keine endgültige Entscheidung treffen, keine verbindliche Rechtsauskunft geben, keine eindeutige Ja-/Nein-Antwort als abschließendes Ergebnis formulieren, kein eigenes Wissen, keine Rechtsprechung, keine Verwaltungspraxis und keine externen Quellen verwenden. Entwickle stattdessen mehrere mögliche Antwortoptionen und stütze jede Option mit wörtlichen Belegen aus dem hochgeladenen Dokument.

Arbeitsregeln. (1) Antworte nicht abschließend mit „Ja" oder „Nein". (2) Formuliere mehrere Lesarten — Option A: Dafür spricht …; Option B: Dagegen spricht …; Option C: Unklar bleibt …; Option D: Eine alternative Lesart wäre … (3) Jede Option mit mindestens einem wörtlichen Zitat aus dem hochgeladenen Dokument. (4) Genaue Fundstelle pro Zitat — Dokumentname, Paragraph/Artikel, Absatz, Satz/Nummer/Buchstabe, Seitenzahl falls verfügbar. (5) Trenne streng wörtliches Zitat, vorsichtige Paraphrase, mögliche Interpretation, offene Unsicherheit. (6) Verwende vorsichtige Formulierungen („Dafür spricht der Wortlaut …", „Dagegen könnte sprechen …", „Eine mögliche Lesart wäre …", „Unklar bleibt anhand des hochgeladenen Dokuments …", „Für eine endgültige Klärung müsste die zuständige Prüfungsstelle einbezogen werden."). (7) Wenn Informationen aus dem Sachverhalt fehlen, nenne sie ausdrücklich. (8) Wenn das Dokument die Frage nicht ausreichend beantwortet: „Im hochgeladenen Dokument nicht eindeutig belegbar." (9) Wenn verschiedene Stellen in Spannung zueinander stehen, benenne diese Spannung, löse sie aber nicht endgültig auf. (10) Erfinde keine Regeln, Fristen, Zuständigkeiten, Ausnahmen oder Rechtsfolgen.

Antwortformat in Schritt 2. A. *Verständnis der Frage* — kurze neutrale Umformulierung. B. *Relevante Textstellen* — Tabelle mit Spalten Fundstelle · Wörtliches Zitat · Mögliche Bedeutung · Relevanz für die Frage. C. *Mögliche*

Antwortoptionen — pro Option: wörtlicher Beleg, Fundstelle, mögliche Bedeutung, Stärke des Belegs (stark/mittel/schwach), warum nicht endgültig. D. *Vergleich der Optionen* — Tabelle mit Spalten Option · Was spricht dafür · Was spricht dagegen · Zentrale Textstelle · Unsicherheitsgrad. E. *Offene Punkte* — fehlende Sachverhaltsangaben, weitere Dokumente, zuständige Hochschulstelle. F. *Prüfprotokoll gegen Halluzinationen* — kurze Antworten zu: Zitat-Beleg pro Option? Unbelegte Aussagen vermieden? Endgültige Entscheidung vermieden? Nur Doku-Inhalt verwendet? Lücken sichtbar gemacht?

Abschluss. Schließe mit: „Diese Analyse zeigt mögliche Lesarten auf Grundlage des hochgeladenen Dokuments. Sie ersetzt keine verbindliche Auskunft der zuständigen Prüfungsstelle."

Kurze Startformulierung für die Studierenden-Rolle.

Tipp

Bitte lade zuerst deine Prüfungsordnung hoch. Ich werde sie zunächst nur als Arbeitsgrundlage erfassen. Danach stellst du eine konkrete Frage dazu. Ich gebe dann keine endgültige Antwort, sondern zeige mehrere mögliche Lesarten mit wörtlichen Belegen aus der Prüfungsordnung.

Schritt 2 — Dokument hochladen und Schritt 1 prüfen (2 Min). Laden Sie Ihre Prüfungsordnung als PDF hoch. Kontrollieren Sie, ob das Modell sauber in Schritt 1 bleibt — nur Identifikation, keine inhaltliche Antwort, keine Spekulation. Falls das Modell vorprescht: Prompt nachschärfen oder Hinweis ergänzen („Bitte zuerst nur identifizieren, nicht antworten").

Schritt 3 — Konkrete Frage stellen (15 Min). Wählen Sie eine reale Frage zu Ihrer Prüfungsordnung — Wiederholungsprüfungen, Fristen, Krankheit, Anerkennung von Leistungen, Abschlussarbeit, Bestehensregeln. Vier Beispielfragen zur Auswahl: *„Darf ich eine bestandene Prüfung freiwillig wiederholen, um die Note zu verbessern?"* · *„Welche Folgen hat ein Attest, das ich nach der Prüfung nachreiche?"* · *„Wie viele Versuche habe ich, wenn ich die Abschlussarbeit nicht bestehe?"* · *„Können Leistungen aus einem Auslandssemester anerkannt werden, wenn das Modul nicht 1:1 vergleichbar ist?"* Stellen Sie die Frage und werten Sie die Antwort aus.

Schritt 4 — Output gegen die Quellenbindung prüfen (8 Min). Gehen Sie das Prüfprotokoll Punkt für Punkt durch und markieren Sie jede Stelle, an der das Modell aus der Rolle fällt.

Prüffrage	Beleg / Befund
Wurde jede Option mit einem wörtlichen Zitat belegt?	
Sind alle Zitate mit exakter Fundstelle versehen (Paragraph, Absatz, Satz/Nummer)?	
Wurde auf eine abschließende „Ja"-/„Nein"-Antwort verzichtet?	
Verwendet das Modell vorsichtige Formulierungen statt subsumierender Sätze?	
Wurden Spannungen zwischen Stellen benannt statt aufgelöst?	
Gibt es Aussagen ohne Zitat-Beleg?	
Hat das Modell eigenes Wissen oder Rechtsprechung eingeschmuggelt?	

Prüffrage	Beleg / Befund
Wurde die Schlussformel zum nicht-verbindlichen Charakter angefügt?	

Erweiterung für schnelle Studierende. Stellen Sie dieselbe Frage einem zweiten Modell mit identischem Prompt und identischem Dokument. Vergleichen Sie die genannten Fundstellen — welche kommen in beiden Antworten vor (stabile Belege), welche nur in einer (instabile Belege)? Was sagt das über die Verlässlichkeit der KI-Recherche im Rechtskontext?

Think-Pair-Share (4 Min). An welcher Stelle hat das Modell trotz Prompt aus der Rolle gewollt? Welche Frage zur Prüfungsordnung würden Sie besser direkt im Prüfungsamt klären als beim Modell — und warum? Wo zahlt sich der zweistufige Aufbau aus, wo hätten Sie ihn lieber einstufig gehabt? Auf welches berufliche Dokument (Vertrag, Bescheid, Mandantenanfrage) ließe sich dieser zweistufige Prompt morgen direkt übertragen?

6.2. Übung 2 — Tutor-Bot systematisch prüfen

Modus: in-class · Einzelarbeit · **Dauer:** 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum) · **4D-Bezug:** Discern.

Prüfen Sie den Tutor-Bot aus Übung 3 (Workshop 1) mit einer kleinen Test-Suite. Drei bis fünf Test-Fragen mit Soll-Antworten formulieren, den Bot durchlaufen lassen, Ergebnisse als *rot* · *gelb* · *grün* markieren, eine Hypothese ableiten. Notizvorlage mit drei Spalten: *Frage* · *Soll-Antwort* · *Ist-Antwort (rot/gelb/grün).* Formulieren Sie in fünf Minuten drei bis fünf fachliche Test-Fragen mit Soll-Antwort in drei bis fünf Sätzen — Beispiele für Tax: Reverse-Charge-Verfahren, verdeckte Gewinnausschüttung, Organschaft. Audit: Going-Concern, Materiality-Schwellen, ISA 315. Advisory: Bewertungsanlässe, KPI-Definition, Business-Case. Stellen Sie jede Frage einzeln und markieren Sie als *rot* (fehlerhaft, halluziniert), *gelb* (teilweise, vage), *grün* (korrekt, vollständig). Schreiben Sie in drei Sätzen, wo der Bot reproduzierbar versagt und woran es vermutlich liegt: Modell, Prompt oder fehlender Kontext.

Erweiterung für schnelle Studierende. Schlagen Sie ein viertes Bewertungskriterium vor, das speziell für Tax-/Audit-Antworten zentral ist — etwa „nennt einschlägige Norm explizit", „benennt Grenzfälle und Ausnahmen" oder „warnt vor berufsrechtlich riskanten Empfehlungen". Begründen Sie, warum dieses Kriterium für Ihr Lerngebiet wichtig ist.

Think-Pair-Share (4 Min). Welche Ihrer Tests waren am aussagekräftigsten — und welche entpuppten sich als zu leicht oder zu schwer? Welcher Anteil Ihrer Tests wäre auch ohne LLM eine sinnvolle Lernkontrolle?

6.3. Übung 3 — RAG-Suchübung HGB-Prüfungspflicht

Modus: in-class oder Hausaufgabe · Einzel oder Paar · **Dauer:** 45 Min · **4D-Bezug:** Discern und Diligence (insbesondere Deployment Diligence).

Trainieren Sie den kontrollierten Einsatz eines RAG-Systems bei einer juristisch geprägten Aufgabe. Das Modell soll nicht entscheiden, subsumieren oder interpretieren — es wird ausschließlich als **quellengebundene Suchhilfe** genutzt. Anders als beim Tutor-Bot wird die KI hier auf eine enge Hilfsrolle zugeschnitten: Textfinder statt Rechtsgutachter.

Setup: RAG-System in der GWDG Academic Cloud (RAG/Arcana oder vergleichbar), HGB-Text aus offizieller Quelle hochladen (Einstieg: Auszug der §§ 264, 264a, 264d, 267, 267a, 316 HGB; Fortgeschrittene: gesamtes HGB), Temperature auf 0 oder maximal 0,1, ein instruktionstreues Modell wählen. **Keine persönlichen oder vertraulichen Daten in den Fall einbauen.**

Fall: Rheinland Components GmbH. Kapitalgesellschaft mit Sitz in Köln. Geschäftsjahr 2024: Bilanzsumme 7,8 Mio. €, Umsatzerlöse 14,9 Mio. €, Arbeitnehmer im Jahresdurchschnitt 51. Geschäftsjahr 2025: Bilanzsumme 7,6 Mio. €, Umsatzerlöse 15,2 Mio. €, Arbeitnehmer im Jahresdurchschnitt 49. Die Geschäftsführung fragt, welche HGB-Textstellen relevant sein könnten, um zu prüfen, ob der Jahresabschluss und der Lagebericht für das Geschäftsjahr 2025 durch einen Abschlussprüfer geprüft werden müssen.

Studentische Suchfrage (keine Entscheidungsfrage).

Tipp

Welche Textstellen im bereitgestellten HGB-Text könnten relevant sein, um zu prüfen, ob der Jahresabschluss und der Lagebericht der Rheinland Components GmbH für das Geschäftsjahr 2025 prüfungspflichtig sind? Bitte gib nur wörtliche Textstellen mit Fundstellen zurück. Keine Interpretation, keine Subsumtion, keine rechtliche Bewertung.

System-Prompt zum Kopieren.

Tipp

Rolle. Du bist eine quellengebundene Suchhilfe für Gesetzestexte im Bereich Wirtschaftsprüfung und Rechnungslegung. Deine Aufgabe ist ausschließlich, passende Textstellen aus dem bereitgestellten Gesetzestext zu finden und wörtlich wiederzugeben.

Verbote. Du darfst keine rechtliche Bewertung vornehmen, keine Subsumtion durchführen, nicht interpretieren, keine abschließende Antwort geben, kein eigenes Wissen verwenden. Du darfst nur mit dem bereitgestellten Material arbeiten. Wenn eine passende Textstelle nicht im bereitgestellten Material enthalten ist, schreibe: „Im bereitgestellten Material wurde keine passende Textstelle gefunden."

Aufgabe. Suche im bereitgestellten Gesetzestext nach Textstellen, die für die folgende Frage relevant sein könnten: [HIER FRAGE EINFÜGEN].

Arbeitsregeln. (1) Gib ausschließlich Textstellen zurück, die im bereitgestellten Material enthalten sind. (2) Zitiere die Textstellen wörtlich. (3) Gib zu jeder Textstelle eine genaue Fundstelle an: Gesetz · Paragraph · Absatz · Satz / Nummer / Buchstabe · Seitenzahl oder Chunk-ID, falls verfügbar. (4) Erkläre nicht, was die Textstelle rechtlich bedeutet. (5) Verwende keine Formulierungen wie „Das bedeutet …", „Daraus folgt …", „Die Gesellschaft ist prüfungspflichtig …". (6) Eine sehr kurze neutrale Suchbegründung ist erlaubt: „Diese Stelle enthält Begriffe zur Prüfungspflicht", „Diese Stelle enthält Größenmerkmale". (7) Wenn mehrere Textstellen relevant sein könnten, ordne sie thematisch.

Antwortformat. A. Gefundene Textstellen. B. Möglicherweise zusätzlich relevante Stellen. C. Nicht gefundene Punkte. D. Prüfhinweis: „Diese Ausgabe ist nur eine Such- und Zitierhilfe. Sie enthält keine rechtliche Bewertung und keine

verbindliche Antwort.”

Kurzprompt-Variante.

Tipp

Finde im bereitgestellten HGB-Text ausschließlich wörtliche Textstellen zu: (1) Prüfungspflicht von Kapitalgesellschaften, (2) kleine Kapitalgesellschaften, (3) mittelgroße Kapitalgesellschaften, (4) Größenmerkmale: Bilanzsumme, Umsatzerlöse, Arbeitnehmer, (5) Bedeutung mehrerer Abschlussstichtage oder Geschäftsjahre. Tabelle: Thema · Fundstelle · Wörtliches Zitat. Keine Interpretation. Keine rechtliche Bewertung. Keine Antwort auf den Fall.

Vorgehen. HGB-Auszug oder gesamtes HGB in das RAG-System hochladen. Suchprompt oder Kurzprompt verwenden. Antwort speichern. Prüfen, ob die Ausgabe wirklich nur Textstellen enthält und keine Interpretation vornimmt. Markieren: mindestens eine starke Fundstelle, eine möglicherweise zusätzliche Fundstelle, eine vermutlich übersehene Stelle.

Selbstprüfung der Modellantwort.

Prüffrage	Ja / Nein
Enthält die Antwort ausschließlich wörtliche Textstellen?	
Sind alle Zitate mit genauer Fundstelle versehen?	
Wurden § 267 HGB und § 316 HGB gefunden?	
Wurde eine Stelle zur Betrachtung mehrerer Abschlussstichtage gefunden?	
Verzichtet die Antwort auf Formulierungen wie „daraus folgt” oder „ist prüfungspflichtig”?	
Sind die Suchbegründungen neutral und kurz?	
Gibt es offensichtlich fehlende Normbereiche?	
Gibt es Zitate, die nicht aus dem bereitgestellten Material stammen können?	

Verbotene und erlaubte Formulierungen.

Nicht verwenden	Stattdessen verwenden
„Daraus folgt, dass die GmbH prüfungspflichtig ist.”	„Diese Stelle enthält Begriffe zur Prüfungspflicht.”
„Die Schwellenwerte sind überschritten.”	„Diese Stelle enthält Größenmerkmale und Schwellenwerte.”
„Die Gesellschaft ist nicht klein.”	„Diese Stelle enthält die Umschreibung kleiner Kapitalgesellschaften.”
„§ 316 HGB ist hier anwendbar.”	„Diese Stelle nennt die Pflicht zur Prüfung.”
„Nach dem HGB gilt …”	„Im bereitgestellten Text steht wörtlich …”

Erweiterung für schnelle Studierende.

Variante	Durchführung
Promptvergleich	Eine Gruppe nutzt den langen Suchprompt, eine den Kurzprompt.
Dokumentvergleich	Ein Lauf mit Auszug, ein Lauf mit gesamtem HGB.
Modellvergleich	Zwei Modelle mit gleichem RAG-Kontext und Prompt.
Fehlerjagd	Eine Antwort mit eingebauten Interpretationen, Studierende markieren unerlaubte Formulierungen.

Reflexionsfragen. Hat das Modell echte Sucharbeit geleistet oder doch interpretiert? Welche Normstellen wurden gefunden, welche möglicherweise übersehen? Wie stark hängt die Ausgabe vom hochgeladenen Dokumentausschnitt ab? Welche Risiken bleiben trotz RAG bestehen?

6.4. Übung 4 — Diligence-Risiko-Recherche im Best-of-N

Modus: in-class · Einzel oder Paar · **Dauer:** 22 Min (15 Min Übung + 4 Min TPS + 3 Min Plenum) · **4D-Bezug:** Diligence, mindestens eine der drei Säulen.

Aus der offiziellen Definition einer Diligence-Säule lässt sich ein starkes KI-Modell mehrere reale oder realistisch konstruierte Anwendungsfälle aus Tax, Audit oder Advisory generieren, in denen genau dieser Aspekt schief gegangen ist. Aus den generierten Fällen wählen Sie einen aus, der das Risiko besonders eindrücklich verdeutlicht — ein praktisches Beispiel für **Best-of-N**. Setup: ein starkes KI-Modell mit Web-Search-Tool (Claude Pro, ChatGPT Plus, Gemini Advanced oder Perplexity Pro), die drei Diligence-Definitionen aus Block K. Wählen Sie eine Säule — Creation, Transparency oder Deployment.

Schritt 1 — Definition als Input (2 Min). Kopieren Sie die vollständige Definition der gewählten Diligence-Säule (alle drei Bullet-Punkte plus Ziel-Satz) in einen neuen Chat. Hängen Sie die Aufgabenstellung an:

> Generieren Sie mir fünf reale oder realistisch konstruierte Anwendungsfälle aus dem Bereich Tax, Audit oder Advisory, in denen mindestens einer der oben genannten Aspekte schief gegangen ist. Pro Fall: ein Satz Sachverhalt, ein Satz Schaden, ein Satz konkret verletzter Aspekt aus der Definition. Verwenden Sie keine erfundenen Namen realer Personen oder Firmen.

Schritt 2 — Auswahl im Best-of-N (8 Min). Lesen Sie die fünf Fälle. Wählen Sie einen aus, der das Risiko besonders eindrücklich verdeutlicht. Auswahlkriterien: Plausibilität, Schwere, Übertragbarkeit.

Schritt 3 — Schutzmaßnahme ableiten (5 Min). Schreiben Sie für den ausgewählten Fall in drei Sätzen: Was ist konkret schief gegangen? Welcher Punkt aus der Diligence-Definition wurde verletzt? Welche konkrete Wenn-Dann-Schutzmaßnahme hätte den Fall verhindert? Diese Regel ist Saatkorn für die Personal AI Policy in Hausaufgabe 7.

Erweiterung für schnelle Studierende. Lassen Sie sich mit demselben Setup fünf weitere Fälle generieren — diesmal mit der zusätzlichen Anweisung „wählen Sie Fälle, die ich aus dem ersten Lauf nicht erwartet hätte". Welche Risiken kommen erst im zweiten Lauf zur Sprache? Was sagt das über die Robustheit des Best-of-N-Vorgehens aus?

Think-Pair-Share (4 Min). Welcher der fünf Fälle wäre fast Ihre Auswahl gewesen — und warum haben Sie sich für einen anderen entschieden? Wie verlässlich sind die generierten Fälle? Gibt es welche, die Sie bei Quellenrecherche entkräften müssten?

6.5. Übung 5 — Prozessmodellierung mit Mermaid

Modus: in-class · Einzelarbeit · **Dauer:** 18 Min · **4D-Bezug:** Vorbereitung für Delegate (Process-Aktivitäten klar identifizieren).

Lernen Sie Prozessmodellierung am niedrigschwelligen Werkzeug **Mermaid Live Editor** kennen (mermaid.live). Mermaid ist eine textbasierte Notation für Diagramme — Sie schreiben den Prozess in einer einfachen Syntax, der Editor zeichnet ihn automatisch. Vorteil für den Einstieg: keine Maus-Klickerei, kein Tool-Setup, sofortige visuelle Rückmeldung.

Sachverhalt: **Eingang einer Mandantenanfrage in einer Steuerkanzlei**. Anfrage trifft per E-Mail ein, wird auf Mandantenbezug geprüft, an einen Sachbearbeiter zugewiesen, dieser klärt offene Punkte, der Partner gibt frei oder verlangt Überarbeitung.

Vorgehen: Öffnen Sie mermaid.live. Löschen Sie den Beispielcode. Kopieren Sie folgendes Grundgerüst hinein und passen Sie es an Ihren Sachverhalt an:

```
graph TD
    A[Anfrage trifft ein] --> B{Mandant bekannt?}
    B -- Ja --> C[Sachbearbeiter zuweisen]
    B -- Nein --> D[Mandant anlegen]
    D --> C
    C --> E[Offene Punkte klären]
    E --> F{Freigabe durch Partner?}
    F -- Ja --> G[Antwort an Mandant]
    F -- Nein --> E
    G --> H[Vorgang archivieren]
```

Schritte: (1) Knoten umbenennen, sodass mindestens fünf Aktivitäten und ein zweites Gateway entstehen. (2) Eine Aktivität als P (Process Automation), eine als C (Cognitive Automation) markieren — etwa durch zusätzliche Knoten oder Notiz im Begleittext. (3) Das gerenderte Diagramm als PNG exportieren (rechte Seite, Download-Knopf).

Erweiterung für schnelle Studierende. Modellieren Sie eine zweite Variante des Prozesses, in der die initiale Mandantenprüfung an einen KI-Tutor delegiert wird. Welche Aktivitäten kommen hinzu, welche werden überflüssig?

Think-Pair-Share (3 Min). Welche Stelle Ihres Diagramms ließ sich in Mermaid leichter zeichnen als auf Papier? Wo stieß die Mermaid-Notation an Grenzen?

6.6. Übung 6 — BPMN-Modellierung in Signavio

Modus: in-class · Einzel oder Paar · **Dauer:** 35 Min · **4D-Bezug:** Delegate / Brücke zu *Process vs. Cognitive Automation.*

Vertiefen Sie die Prozessmodellierung mit einem professionellen BPMN-Werkzeug. Anders als bei Mermaid sind in Signavio die Symbole standardisiert (BPMN 2.0.2), Zuständigkeiten lassen

sich über Swimlanes sichtbar machen, und das Modell ist auditierbar. Setup: Signavio Academic Edition (kostenfreier Account mit Hochschul-E-Mail; sollte bereits aus der Hausaufgabe nach Workshop 1 angelegt sein). Als Alternative: bpmn.io.

Sachverhalt: **Rechnungseingang in einer mittelgroßen Steuerkanzlei** — Posteingang, Erfassung, sachliche Prüfung, Freigabe, Buchung, Zahlung. Reicher als die Mandantenanfrage in Übung 5.

Vorgehen: Nach Login in Signavio einen neuen Diagrammtyp *BPMN 2.0* anlegen. Mindestens zwei Rollen identifizieren und als Swimlanes setzen (z. B. *Sachbearbeiter* und *Partner*). Mindestens fünf Aktivitäten platzieren und mit Sequenzflüssen verbinden. Mindestens einen XOR-Entscheidungspunkt einsetzen. Jede Aktivität mit *P* (Process Automation), *C* (Cognitive Automation) oder *H* (Hybrid) markieren und in einem Satz pro Aktivität begründen. Diagramm als PNG oder PDF exportieren.

Erweiterung für schnelle Studierende. Eine Aktivität, die heute *H* (Hybrid) ist, in zwei aufeinanderfolgende Aktivitäten zerlegen — eine reine Process-Aktivität, eine reine Cognitive-Aktivität. Hinweis: Das ist der Hauptmechanismus, mit dem reale Workflows automatisierbar werden.

Think-Pair-Share (4 Min). Welche Stelle aus dem Mermaid-Diagramm (Übung 5) hätte in Signavio anders ausgesehen? Wo wirkt BPMN überdimensioniert, wo zahlt sich die strikte Notation aus?

6.7. Übung 7 — UiPath Mandanten-Anschreiben

Modus: in-class · Einzelarbeit · **Dauer:** 22 Min · **4D-Bezug:** Delegate / praktischer Erstkontakt mit Process Automation an einem realistischen Tax-/Audit-Szenario.

Lernen Sie die Grundstruktur von **UiPath Cloud Studio Web** an einer Aufgabe kennen, die in jeder Kanzlei vorkommt: einen **Serienbrief-Bot**, der die Adressdaten eines Mandanten aus einer Excel-Liste in ein Word-Anschreiben einsetzt und das fertige Dokument speichert. Setup: Account bei UiPath Cloud (sollte aus der Hausaufgabe nach Workshop 1 vorhanden sein). Nach Login oben links *Studio Web* öffnen.

Material. Im Daten-Ordner finden Sie zwei vorbereitete Dateien — eine Excel-Stammdatenliste `mandantenstammdaten.xlsx` mit den Spalten *Anrede* · *Vorname* · *Nachname* · *Firma* · *Straße* · *PLZ* · *Ort* und eine Word-Vorlage `anschreiben-vorlage.docx` mit den Platzhaltern `{Anrede}`, `{Vorname}`, `{Nachname}`, `{Firma}`, `{Strasse}`, `{PLZ}`, `{Ort}`. Beide Dateien sind in Ihren UiPath-Cloud-Workspace hochzuladen (oben rechts *Upload*).

Vorgehen — der Bot in sechs Schritten:

1. **Neues Projekt anlegen** — *Create new project*, Typ *Process*, Name „Mandanten-Anschreiben".
2. **Excel-Datei einbinden** — Activity *Use Excel File*, Pfad auf die hochgeladene `mandantenstammdaten.xlsx` setzen, Referenznamen `Stammdaten` vergeben.
3. **Adressdaten lesen** — innerhalb des Excel-Scopes mit *Read Cell* die ersten sieben Felder der Zeile 2 lesen und je einer String-Variablen zuweisen (`Anrede`, `Vorname`, `Nachname`, `Firma`, `Strasse`, `PLZ`, `Ort`). Wer mag, verwendet stattdessen *Read Range* und greift später per Index zu.
4. **Word-Vorlage öffnen** — Activity *Use Word File*, Pfad auf `anschreiben-vorlage.docx`, Option *Make a copy* aktivieren und Ausgabepfad auf `Anschreiben-{Nachname}.docx` setzen — so wird die Vorlage nicht überschrieben.

5. **Platzhalter ersetzen** — sieben *Replace Text in Document*-Activities einfügen, je ein Platzhalter (`{Anrede}` etc.) durch die korrespondierende Variable austauschen.
6. **Bot ausführen** — *Run* klicken, das erzeugte Anschreiben im Workspace öffnen und prüfen, ob alle Platzhalter sauber ersetzt sind.

Anschließend: drei Stichworte ins Notizbuch — wo war die Oberfläche intuitiver als erwartet, wo zäher, welche Stelle wäre bei einer echten Mandantenliste mit 200 Zeilen die kritischste?

Erweiterung für schnelle Studierende. Bauen Sie eine *For Each Row*-Schleife um die Schritte 3 bis 5, sodass der Bot alle Zeilen der Excel-Liste verarbeitet und für jeden Mandanten eine eigene Datei `Anschreiben-{Nachname}.docx` schreibt. Beobachten Sie, wie sich der Aufwand pro Mandant gegen Null bewegt — und wo trotzdem ein Mensch prüfen muss, bevor das Schreiben rausgeht.

Think-Pair-Share (3 Min). An welcher Stelle dieses Bots würden Sie *Cognitive Automation* sinnvoll ergänzen — etwa für eine individuelle Anrede oder einen mandantenspezifischen Hinweis? Welche Diligence-Pflicht (Creation, Transparency, Deployment) trifft den Versand solcher Serienbriefe am stärksten?

6.8. Vertiefung (optional in-class oder als Hausaufgabe)

Wer die sechs Übungen früh durchhat oder eine zusätzliche Vertiefung sucht, hat zwei Optionen.

Integrierter 4D-Use-Case mit Diligence-Schwerpunkt — eine 40-Minuten-Übung am Tax-Beispiel „grenzüberschreitende Dienstleistung", in der alle vier 4D-Kompetenzen mit Schwerpunkt Diligence durchlaufen werden, inklusive Audit-Trail und Mandantenkommunikation. Diese Übung ist gleichzeitig die Vorlage für die zwei Hausaufgaben *Dilemma der Mitte* und *Personal AI Policy*. Beschreibung folgt unter Übung 8.

UiPath-Vertiefung — Excel-zu-PDF-Bot — eine komplexere UiPath-Aufgabe, in der Sie einen vorbereiteten Bot lesen, ausführen und anpassen. Diese Übung ist als Hausaufgabe nach Workshop 2 konzipiert; Details siehe Hausaufgaben-Seite.

6.9. Übung 8 — Integrierter 4D-Use-Case (optional in-class)

Modus: in-class · Paar oder Dreierteam · **Dauer:** 40 Min · **4D-Bezug:** alle vier Kompetenzen, Schwerpunkt Diligence.

Abbildung 6.1.: Centaur-Modell — Mensch und KI als verbundenes, aber unterscheidbares Gespann; die Verantwortung bleibt sichtbar beim Reiter. Adaptiert nach Mollick (2024).

Führen Sie einen realistischen Sachverhalt in einem Durchgang durch alle vier 4D-Kompetenzen — mit besonderem Gewicht auf der Diligence-Dokumentation. Diese Übung ist die direkte Vorlage für die beiden Hausaufgaben *Dilemma der Mitte* und *Personal AI Policy*.

Sachverhalt (fiktiv, kein Mandantenbezug): Eine deutsche IT-Beratung erbringt eine Cloud-Migrations-Beratung im Wert von 80.000 € für einen Kunden mit Sitz in Boston, USA. Frage des Mandanten: Ist die Leistung in Deutschland umsatzsteuerbar — und wenn ja, mit welcher Steuerklasse?

Ablauf in vier Stationen. *Delegate (5 Min)*: Welches Modell mit welcher Harness? Welche Teilaufgaben behalten Sie als Berufsträger:in selbst? *Describe (5 Min)*: Prompt nach RTF oder CREATE strukturieren, absenden. *Discern (10 Min)*: Faithfulness prüfen, zitierte Quellen verifizieren, Honey-Pot setzen, Output als rot/gelb/grün bewerten. *Diligence (20 Min)*: Audit-Trail ausfüllen, Mandantenkommunikation in fünf Sätzen formulieren (EU AI Act Art. 50), Verantwortungs-Schluss-Satz schreiben, Faustregel für die Personal AI Policy notieren.

Erweiterung für schnelle Studierende. Übersetzen Sie den vollständigen Audit-Trail in eine Personal-AI-Policy-Regel als Wenn-Dann-Aussage. Beispiel: „*Wenn* der Sachverhalt einen US-Mandanten betrifft, *dann* verifiziere ich mindestens zwei Quellen aus deutschen Quellen und keine reine LLM-Antwort." Diese Regel ist Saatkorn für die Hausaufgabe *Personal AI Policy*.

6.10. Quellen

Dakan & Feller (2025) · Object Management Group (2014) · M. Lacity & Willcocks (2021) · Willcocks & Lacity (2024) · M. C. Lacity & Willcocks (2016) · Coombs et al. (2020) · Davenport & Ronanki (2018) · Es et al. (2024) · Willison (2025)

Teil III.

Teil 3 — Hausaufgaben & Wissensbasis

7. Hausaufgaben nach Workshop 2

UiPath-Vertiefung · Dilemma der Mitte · Personal AI Policy

7.0.0.1. Nach den Hausaufgaben können Sie

- einen UiPath-Bot lesen, ausführen und an einer Stelle anpassen,
- ein Diligence-Dilemma argumentativ einordnen und begründet Stellung beziehen,
- eine persönliche AI Policy formulieren, die die vier Kompetenzen in eigene Praxisregeln übersetzt.

7.1. UiPath-Vertiefung — Excel-zu-PDF-Bot

Hausaufgabe nach Workshop 2 · Screenshot-Dokumentation und kurzer Beschreibungstext · 4D-Bezug: Delegate.

Öffnen Sie in UiPath Cloud Studio Web den vorbereiteten Bot (`code/uipath-excel-zu-pdf-template.xaml`), starten Sie ihn mit der Beispiel-Excel (`data/eingangsrechnungen-beispiel.xlsx`) und prüfen Sie das Output-PDF. Passen Sie dann eine Aktivität an — Schwellwert für die sachliche Prüfung ändern, ein zusätzliches Prüfkriterium ergänzen oder eine zweite Output-Datei hinzufügen. Vorher/Nachher mit zwei Screenshots dokumentieren plus fünf Sätze Beschreibung der Änderung und ihrer Wirkung.

Wer mehr will: Schlagen Sie eine Stelle vor, an der Cognitive Automation den Bot ergänzen könnte (LLM-Aufruf, OCR-Schicht). Welche Activity käme hinzu, welche neue Fehlerquelle entstünde?

7.2. Übung 6 — Das Dilemma der Mitte

Positionspapier 500–800 Wörter, PDF, APA, mit Quellenverzeichnis · 4D-Bezug: Diligence.

Beschreiben Sie ein konkretes *Dilemma der Mitte* — eine Situation aus Tax, Audit oder Advisory, in der KI-Einsatz weder klar verboten noch klar geboten ist — und beziehen Sie begründet Stellung. Realistischer, nicht-mandantenbezogener Sachverhalt. Schildern Sie das Dilemma in drei bis fünf Sätzen, listen Sie drei Argumente für und drei dagegen, beziehen Sie eine begründete Position mit zwei Quellen, und leiten Sie eine eigene Diligence-Regel ab.

Wer mehr will: Formulieren Sie eine Steelman-Version der Gegenposition. Wenn Ihre Begründung gegen die Steelman noch standhält, ist sie tragfähig.

7.3. Übung 7 — Personal AI Policy

Capstone-Hausaufgabe · 1–3 Seiten, mit Datum, Versionsnummer, Querverweisen zu mindestens drei Übungen · 4D-Bezug: integrativ.

Formulieren Sie eine persönliche AI Policy, die festlegt, wann, wie und mit welchen Schutzmechanismen Sie KI in Ihrer Berufspraxis einsetzen. Adressat: Sie selbst plus eine fiktive Aufsichtsperson. Material aus allen Workshop-Übungen plus Berufsrecht (siehe Wissensbasis · Recht und Berufsstand). Sieben Sektionen: *Geltungsbereich* · *Delegation-Regeln* · *Description-Regeln* · *Discernment-Regeln* · *Diligence-Regeln* · *Eskalationspfade* · *Review-Rhythmus*.

Wer mehr will: Mindestens drei Regeln so umformulieren, dass sie konkret eine Übung aus Tag 1 oder 2 referenzieren — dadurch wird aus der generischen Compliance-Liste ein persönliches Dokument.

8. Wissensbasis

Nachschlagewerk zu LLMs, Modellen, Prompts, Tools, BPMN, RPA und Berufsrecht

8.0.0.1. Was Sie in dieser Wissensbasis finden

- Kompakte Nachschlage-Sektionen zu allen Themen, die in den Workshops 1 und 2 verwendet werden,
- Lay-Erklärung jedes Fachbegriffs **vor** dem technischen Detail,
- Verweise auf die Originalliteratur und zu den passenden Workshop-Stellen.

Diese Seite ist absichtlich als ein einziges, durchsuchbares Dokument angelegt. Stand: 12.05.2026.

Vertiefung — Grundlagen und Prompt-Beispiele

Ein eigenes Grundlagenskript ergänzt diese Wissensbasis um eine ausführliche Einführung in die Funktionsweise von Sprachmodellen und eine umfangreiche Sammlung kommentierter Prompt-Beispiele zum Lernen und Adaptieren.

- **Grundlagen Sprachmodelle und „How to speak"** — Kapitel 2 von *GenAI in der Lehre* (Bartnik, 2026)
- **Prompt-Beispiele-Sammlung** — Kapitel 7, Appendix 2 mit kommentierten Vorlagen für Tutor-Bots, Erklärbots, Constraints und mehr
- **Komplettes Skript** — genai4teaching.github.io

8.1. AI Fluency Framework — die vier Kompetenzen im Überblick

Die folgenden Definitionen folgen dem *Framework for AI Fluency — Practical Overview Document, Version 1.1* von Dakan & Feller (2025). Übersetzung aus dem englischen Original. Lizenz des Originals: CC BY-NC-ND 4.0.

Was ist AI Fluency?

AI Fluency ist die Fähigkeit, **effektiv**, **effizient**, **ethisch** und **sicher** innerhalb der entstehenden Modalitäten der Mensch-KI-Interaktion zu arbeiten (Dakan & Feller, 2025). Das Framework benennt vier vernetzte Kernkompetenzen — die *4 Ds* — sowie drei Modalitäten, in denen Mensch und KI zusammenwirken.

8.1.1. Drei Modalitäten der Mensch-KI-Interaktion

Modalität 1 — Automation (KI führt eine vom Menschen definierte Aufgabe aus). KI erledigt Aufgaben eigenständig, aber auf der Grundlage direkter menschlicher Anweisungen (etwa als Antwort auf einen Prompt). Besonders geeignet für repetitive, zeitintensive oder datenintensive Aufgaben. Erfordert klare Aufgabendefinition und Qualitätskontrolle. Beispiele: E-Mails, Zusammenfassungen, Social-Media-Posts, einfaches Coding.

Modalität 2 — Augmentation (KI und Mensch erledigen die Aufgabe gemeinsam). KI und Mensch definieren und führen Aufgaben iterativ gemeinsam aus und arbeiten kollaborativ auf ein Endziel hin. Schwerpunkt liegt auf der Stärkung menschlicher Kreativität durch einen KI-Denkpartner, nicht auf Ersetzung. Beispiele: Geschichten und Essays schreiben, Forschungsarbeiten, komplexes Coding.

Modalität 3 — Agency (Mensch konfiguriert KI für eigenständige zukünftige Tätigkeiten). Der Mensch konfiguriert die KI, damit sie zukünftige Aufgaben — auch für andere Nutzer:innen — selbstständig erledigt. Diese Modalität definiert nicht eine konkrete Aufgabe, sondern Eigenschaften und Verhalten einer KI. Erfordert ein tiefes Verständnis von Fähigkeiten und Grenzen. Beispiele: interaktive Spielcharaktere, Tutoren, Chatbots.

Mensch-KI-Interaktionen überbrücken häufig mehrere Modalitäten gleichzeitig; in der Praxis wechseln Anwendende selbst innerhalb eines Projekts zwischen ihnen.

8.1.2. Delegation — Schaffende Vision und Auswahl der richtigen KI-Werkzeuge

Delegation bezeichnet die Fähigkeit zu erkennen, *wann* und *wie* KI-Werkzeuge und -Modalitäten in kreativen und problemlösenden Prozessen wirksam eingesetzt werden. Sie umfasst das Verständnis der Möglichkeiten und Grenzen verschiedener KI-Technologien und das informierte Entscheiden, wann KI für Automation, Augmentation oder Agency genutzt wird.

Ziel- und Aufgabenbewusstsein (Goal and Task Awareness).

- Eine wirksame Zielvorstellung für ein Projekt entwerfen.
- Die Natur und die Anforderungen der Aufgabe(n) auf dem Weg zum definierten Ziel verstehen.
- Eine Aufgabe in KI-, Mensch- und kollaborative Anteile analysieren und zerlegen können.
- Voraussetzung für die effektive Integration von KI in kreative Arbeitsabläufe.

Plattformbewusstsein (Platform Awareness).

- Möglichkeiten und Grenzen aktueller KI-Werkzeuge kennen.
- Verschiedene KI-Plattformen und ihre spezifischen Stärken und Grenzen mit Blick auf das Projektziel kennen.
- KI-Werkzeuge nach Projektanforderungen, Budget, operativen und regulatorischen Bedürfnissen bewerten können.
- Voraussetzung für die Auswahl der optimalen KI-Werkzeuge für spezifische Aufgaben.

Aufgabendelegation (Task Delegation).

- KI- und menschliche Fähigkeiten so ausbalancieren, dass die kreative Vision optimal umgesetzt wird.
- Die Eigenschaften der drei Modalitäten (Automation, Augmentation, Agency) verstehen.
- Projektaufgaben optimal an Mensch und KI-Werkzeuge vergeben können.
- Voraussetzung für die erfolgreiche Zusammenarbeit zwischen Mensch und KI in kreativen Prozessen.

8.1.3. Description — Vision und Aufgaben so beschreiben, dass nützliche KI-Verhaltensweisen entstehen

Description umfasst die Fähigkeiten, Ideen, Anforderungen, Constraints und andere Aspekte kreativer Vorstellungen wirksam an KI-Systeme zu kommunizieren. Sie beinhaltet das Verfassen klarer, präziser und gut strukturierter Prompts (mit einem breiten Repertoire an Prompting-Techniken) und weiterer Elemente, die KI-Werkzeuge zu gewünschten Verhaltensweisen und Ergebnissen führen.

Produktbeschreibung (Product Description).

- Prompten, um das gewünschte Ergebnis zu definieren.
- Gewünschte Eigenschaften, Merkmale und Qualitäten des finalen KI-generierten Outputs klar artikulieren können.
- Eine kreative Vision in explizite, KI-verständliche Begriffe übersetzen können.
- Zentral, um KI-Werkzeuge zu Ergebnissen zu führen, die mit den Absichten der Schaffenden übereinstimmen.

Prozessbeschreibung (Process Description).

- Dialogisches Prompten für effektive iterative Zusammenarbeit.
- In einen dynamischen Hin-und-Her-Austausch mit KI-Werkzeugen treten können.
- Komplexe Aufgaben in eine Reihe kleinerer, handhabbarer Prompts zerlegen können.
- Wesentlich, um KI durch mehrstufige kreative Prozesse zu führen, im Einklang mit den menschlichen Mitwirkenden.

Verhaltensbeschreibung (Performance Description).

- Anweisendes Prompten, das zukünftige KI-Verhaltensweisen festlegt und ein positives Nutzungserlebnis ermöglicht.
- Definieren können, wie KI-erzeugte Inhalte oder Systeme sich verhalten oder mit der Welt interagieren sollen.
- Nutzungsbedürfnisse antizipieren und in Leitlinien für KI-Verhalten übersetzen können.
- Entscheidend, um zukünftige agentische KI-Verhaltensweisen zu ermöglichen, die mit Werten und Vorstellungen des Menschen übereinstimmen.

8.1.4. Discernment — Treffende Beurteilung des Nutzens von KI-Ergebnissen

Discernment umfasst die kritische Beurteilung KI-generierter Outputs hinsichtlich Qualität, Relevanz, möglicher Verzerrungen und weiterer wesentlicher Merkmale. Dazu gehört auch die Fähigkeit, den Kollaborationsprozess mit KI-Werkzeugen zu iterieren und zu verfeinern.

Produkt-Beurteilung (Product Discernment).

- Output-Qualität bewerten und Verbesserungsmöglichkeiten identifizieren.
- Qualität, Relevanz und Wirksamkeit KI-generierter Inhalte kritisch beurteilen können.
- Stärken und Schwächen in KI-Outputs erkennen können.
- Zentral für das Aufrechterhalten hoher Standards in KI-gestützter kreativer Arbeit.

Prozess-Beurteilung (Process Discernment).

- Beurteilen, ob die Mensch-KI-Zusammenarbeit produktiv ist und wie sie verbessert werden kann.
- Die Wirksamkeit des Mensch-KI-Kooperationsprozesses evaluieren können.

- Erkennen, welche Aspekte der Mensch-KI-Interaktion am hilfreichsten sind und wo Verbesserungen möglich sind.
- Wesentlich für die Optimierung des KI-Einsatzes in kollaborativer kreativer Arbeit.

Verhaltens-Beurteilung (Performance Discernment).

- Beurteilen, ob eigenständige KI-Verhaltensweisen ein positives Nutzungserlebnis ermöglichen, und wie die KI besser angeleitet werden kann.
- Die Wirksamkeit von KI-Systemen in unabhängigen, nutzerorientierten Szenarien beurteilen können.
- Nutzerfeedback erheben und interpretieren, um beabsichtigte KI-Verhaltensweisen und Erlebnisse zu verfeinern und sicherzustellen.
- Wesentlich für die Gestaltung von Nutzungserlebnissen, die mit den Werten und der Vision des Projekts übereinstimmen.

8.1.5. Diligence — Verantwortung übernehmen und für KI-gestützte Endprodukte einstehen

Diligence bezeichnet den verantwortlichen Umgang mit KI, einschließlich ethischer Überlegungen, Transparenz über den KI-Einsatz und die Übernahme von Verantwortung für die mit KI-Unterstützung erstellten Endprodukte.

Creation Diligence — verantwortlicher Umgang während der Erzeugung.

- Verantwortungsvoller Umgang mit KI-Tools unter Einhaltung ethischer und rechtlicher Best Practices sowie Bewusstsein für Verzerrungen, Mängel, Auswirkungen auf Interessengruppen und andere externe Effekte.
- Verständnis und Anwendung ethischer Grundsätze während des gesamten KI-gestützten kreativen Prozesses.
- Fähigkeit, potenzielle Verzerrungen und ethische Risiken in KI-generierten Inhalten zu erkennen und zu mindern.
- Ziel: Gewährleistung eines verantwortungsvollen und sozialbewussten Einsatzes von KI.

Transparency Diligence — Offenlegung gegenüber Stakeholdern.

- Transparenz und Verantwortlichkeit bei der Verbreitung des Endprodukts.
- Verständnis für die Erwartungen und Normen des Publikums, der Branche und der Rechtsordnung in Bezug auf KI-generierte Inhalte.
- Fähigkeit, die Art der KI-Beteiligung am Prozess klar zu kommunizieren.
- Ziel: Wahrung von Vertrauen und Integrität bei der Verbreitung KI-gestützter Arbeiten.

Deployment Diligence — Verantwortung für Verifikation und Veröffentlichung.

- Verantwortung für die Verifikation von und das Einstehen für KI-gestützte Outputs übernehmen, einschließlich gründlicher Faktenprüfung, Test auf Korrektheit und Validierung von Aussagen.
- Angemessene Sicherheitsprüfungen und Testverfahren vor der Freigabe KI-gestützter Arbeit umsetzen.
- Verstehen, managen und Verantwortung übernehmen für potenzielle Risiken und Auswirkungen veröffentlichter KI-gestützter Inhalte oder Agenten.
- Ziel: Sicherstellung von Qualität, Sicherheit und Verlässlichkeit von Inhalten und Agenten, die durch Mensch-KI-Interaktion entstanden sind.

Querverweise: Workshop 1, Block 2 (Kurzdurchlauf der vier Kompetenzen) · Workshop 2, Block K (Diligence vertieft) · Übungen Workshop 1 (Anwendung von Delegation, Description, Discernment).

Quelle: Dakan, R. & Feller, J. (2025). *Framework for AI Fluency: Practical Overview Document* (V 1.1). CC BY-NC-ND 4.0. https://ringling.libguides.com/ai/framework

8.2. LLM-Grundlagen

> **Was ist ein LLM?**
>
> Ein **Large Language Model** ist ein **statistischer Textgenerator**, der das jeweils nächstwahrscheinliche Wort vorhersagt — nicht mehr und nicht weniger. *Bild:* sehr gut geübter Autovervollständiger, der zu einem Anfang den passenden Folgetext rät. *Technisch:* tiefes neuronales Netz mit Transformer-Architektur, trainiert auf großen Textmengen.

Drei Kernunterscheidungen, die durchgehend in beiden Workshops auftauchen.

8.2.1. Reasoning vs. Instant

Instant-Modelle antworten direkt; **Reasoning-Modelle** denken zwischen Frage und Antwort sichtbar oder unsichtbar nach (Chain-of-Thought). Reasoning lohnt sich bei Mehrschritt-Aufgaben, ist langsamer und teurer.

8.2.2. Werkzeuge und Harness

Harness — der **Werkzeugring** um das LLM herum: Web-Search, Code-Interpreter, Datei-Upload, Memory, Tool-Calling. Die Antwortqualität hängt fast immer mehr vom Harness ab als vom Modell allein (Grootendorst, 2025).

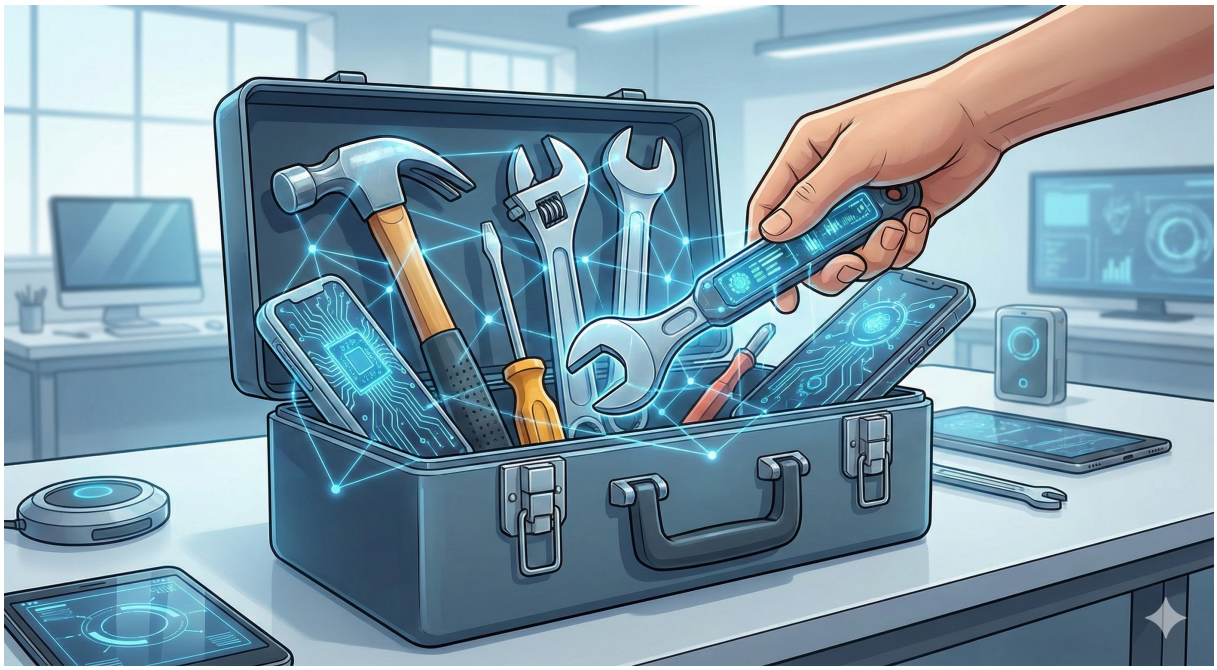


Abbildung 8.1.: Tool-Metapher — ein LLM ohne Harness ist wie ein guter Handwerker ohne Werkzeugkasten; erst die Tools machen aus dem Modell ein System.

8.2.3. Drei Lizenz-Tiers

Free (0 €): begrenzter Kontext, oft kein Web-Search, keine vertraulichen Daten. **Casual** (~20–30 €/Monat): Pro-Tier von Claude, ChatGPT, Gemini. **Professional** (~100–200 €/Monat): Team-/Enterprise-Tiers mit Datenschutzgarantien, längeren Kontexten, mehr Tools. *Faustregel:* Mandantenarbeit nicht im Free-Tier.

Querverweise: Workshop 1, Block 3 · Übung 2.

8.3. Welches Modell?

Schlanke Vergleichsseite analog zu Békes (2026). Stand der Empfehlungen: Mai 2026.

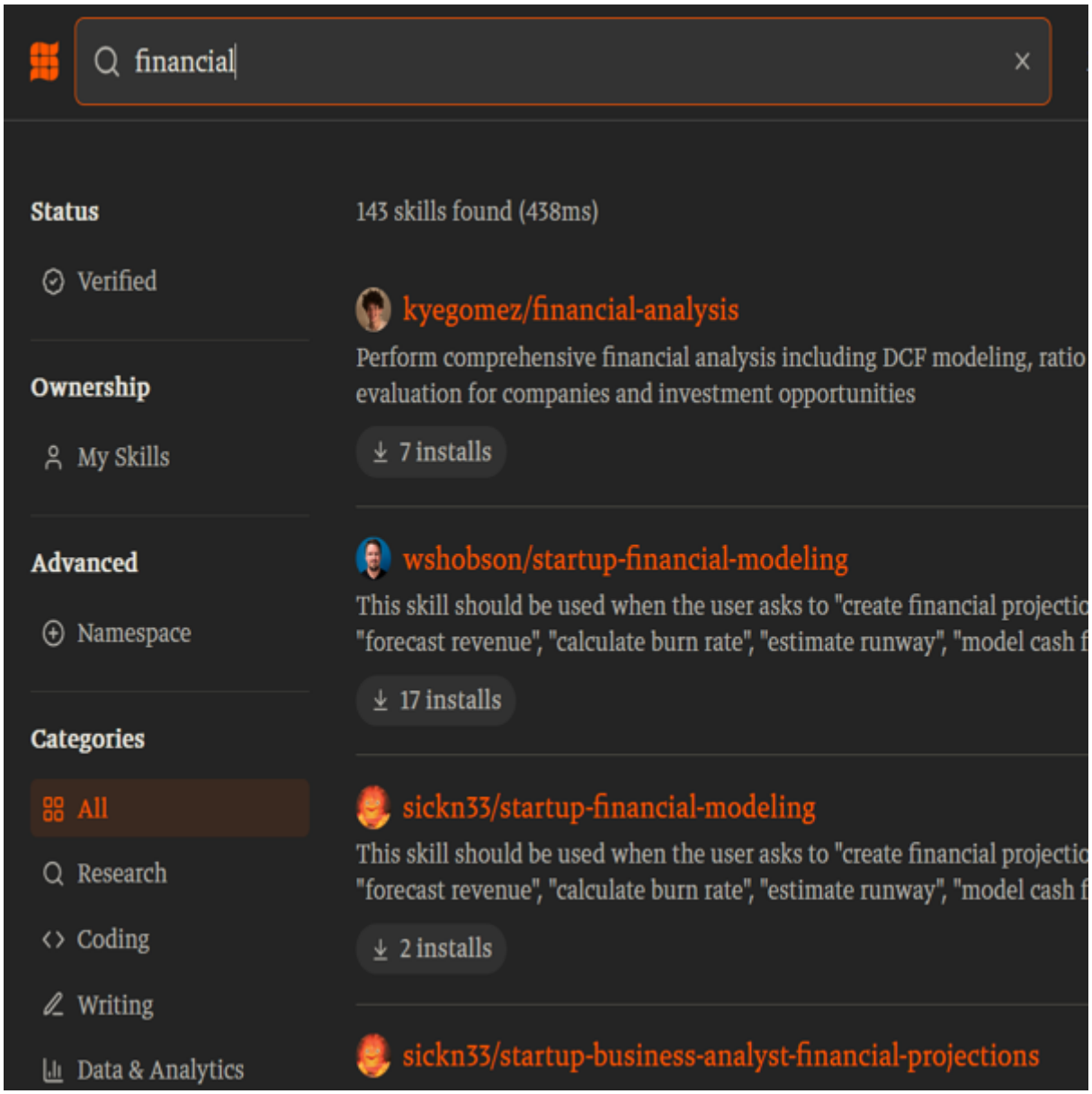


Abbildung 8.2.: Skill-Profil im Finanzbereich — typische Tätigkeitsfelder, in denen sich Modell-Fähigkeiten ablesen und vergleichen lassen.

Querverweise: Workshop 1, Block 3 · Übung 2.

8.4. Prompt-Muster

Sieben Bausteine eines belastbaren Prompts: **Rolle** · **Zielgruppe** · **Aufgabe** · **Tonalität** · **Constraints** · **Format** · **Iteration**. Anker: Project Management Institute (2024).

8.4.1. Iteration als Pflicht

Ein Prompt ist nie fertig. Drei Iterationen sind Mindeststandard, dokumentiert mit *was geändert* · *warum* · *Effekt auf den Output*.

Querverweise: Übung 2 · Übung 3.

8.5. Prompt-Pattern-Katalog nach White et al. (2023)

White et al. (2023) katalogisieren sechzehn wiederverwendbare *Prompt-Patterns* — Bauformen, mit denen sich häufige Probleme im Umgang mit Sprachmodellen systematisch lösen lassen. Die Patterns sind keine konkreten Beispiele, sondern Schablonen, die sich auf jede Domäne übertragen lassen. Die Autoren gruppieren die Patterns in sechs Kategorien.

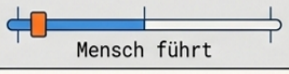
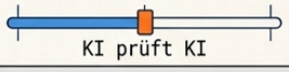
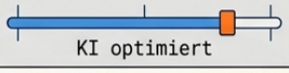
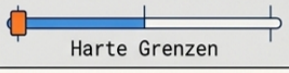
Kategorie	Primäres Ziel	Steuerung (Control)	Kognitiver Aufwand
Output Customization	Daten strukturieren & formatieren		 Niedrig
Error Identification	Halluzinationen verhindern & Fakten sichern		 Mittel
Prompt Improvement	Qualität der Nutzer-Fragen optimieren		 Mittel
Interaction	Dynamische Problemlösung & Dialog		 Hoch
Context Control	Relevanz filtern & Kontext limitieren		 Niedrig

Abbildung 8.3.: Sechs Prompt-Kategorien nach White et al. (2023) — jede Kategorie adressiert ein anderes Problem im Umgang mit Sprachmodellen.

Die sechs Kategorien adressieren je ein anderes Problem im Dialog mit dem Sprachmodell. **Input Semantics** stellt die Verständigung her — eigene Notationen und Kurzschriften, mit denen wiederkehrende Sachverhalte präzise und kompakt eingegeben werden. **Output Customization** formt das Ergebnis — Persona, Format, Struktur und Granularität der Antwort werden vorab festgelegt, damit die Ausgabe direkt anschlussfähig ist. **Error Identification** baut Selbstprüfung ein — das Modell wird verpflichtet, Annahmen offenzulegen, Faktenbehauptungen zu markieren und die eigene Antwort gegenzulesen. **Prompt Improvement** verbessert die Frage selbst — das Modell schlägt präzisere Formulierungen vor, prüft auf fehlende Voraussetzungen und liefert Alternativen statt einer einzelnen Antwort. **Interaction** dreht den Dialog um — statt einer Einbahnstraße entstehen geführte Befragungen, Spielszenen oder iterative Generierungsläufe. **Context Control** schließt die Tür — der zulässige Wissensraum wird auf die mitgelieferten Quellen begrenzt, Trainingswissen ausgeschlossen.

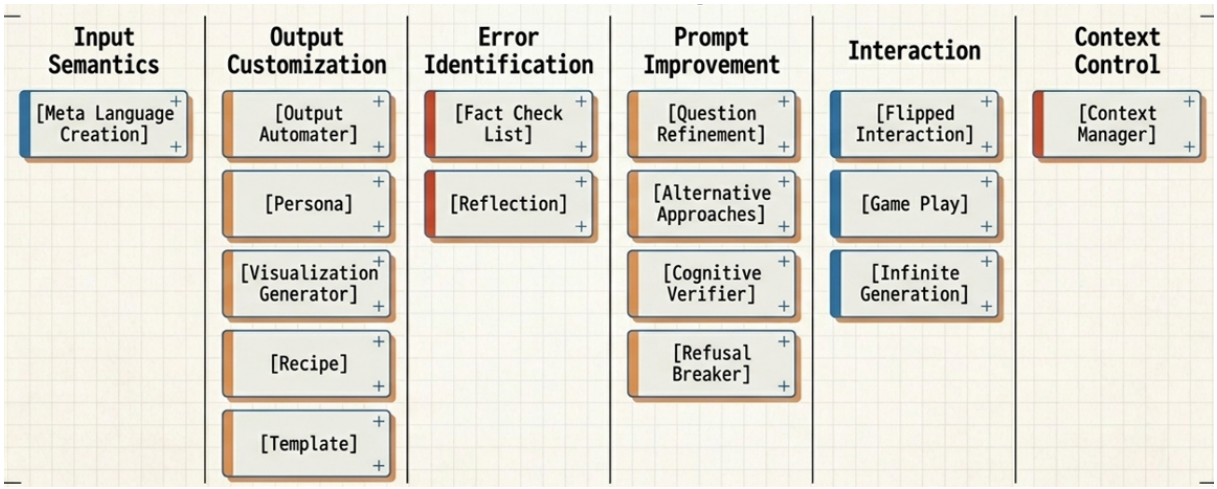


Abbildung 8.4.: Sechs Prompt-Kategorien und die zugeordneten 16 Muster nach White et al. (2023).

Die sechzehn Muster verteilen sich ungleichmäßig auf die Kategorien — *Output Customization* trägt mit fünf Mustern (Output Automater, Persona, Visualization Generator, Recipe, Template) das größte Gewicht, weil die Form der Antwort über ihre Brauchbarkeit entscheidet. *Prompt Improvement* folgt mit vier Mustern (Question Refinement, Alternative Approaches, Cognitive Verifier, Refusal Breaker), *Interaction* mit drei (Flipped Interaction, Game Play, Infinite Generation). *Input Semantics* (Meta Language Creation) und *Context Control* (Context Manager) bestehen aus je einem Muster — sie sind eng umrissen, aber strukturell zentral. *Error Identification* hat zwei (Fact Check List, Reflection). Die folgende Tabelle gibt einen kompakten Überblick.

Kategorie	Patterns
Input Semantics	Meta Language Creation
Output Customization	Output Automater · Persona · Visualization Generator · Recipe · Template
Error Identification	Fact Check List · Reflection
Prompt Improvement	Question Refinement · Alternative Approaches · Cognitive Verifier · Refusal Breaker
Interaction	Flipped Interaction · Game Play · Infinite Generation
Context Control	Context Manager

Pro Pattern folgt erst das allgemeine Prinzip nach White et al., dann zwei Anwendungsbeispiele aus Tax, Audit oder Advisory. Patterns sind kombinierbar — viele Workshop-Übungen nutzen mehrere gleichzeitig.

8.5.1. Meta Language Creation

Prinzip. „Wenn ich X sage, meine ich Y." — Das Modell wird auf eine eigene Kurzschrift trainiert, mit der sich Sachverhalte kompakter beschreiben lassen, als es in Fließtext möglich wäre. Geeignet für wiederkehrende Strukturen mit komplexer Syntax. Vorsicht: pro Session nur eine Meta-Sprache, sonst entstehen Konflikte.

Tax-Beispiel. „Wenn ich `§-Verweis(Norm, Absatz, Satz)` schreibe, beziehe ich mich auf die genannte Norm. Beispiel: `§-Verweis(KStG, 8, 3)` meint § 8 Abs. 3 KStG."

Audit-Beispiel. „Wenn ich `Bef-(Aktivität, Risiko, Materialität)` schreibe, beschreibe ich einen Prüfungsbefund. Beispiel: `Bef-(Bestandsaufnahme, vollständig, niedrig)` meint einen Befund zur Bestandsaufnahme ohne Materialitäts-Auffälligkeit."

8.5.2. Output Automater

Prinzip. „Wenn deine Antwort einen ausführbaren Ablauf nahelegt, liefere mir zusätzlich ein Skript oder eine Automatisierungs-Artefakt, das diese Schritte ausführt." Spart manuelles Übertragen mehrstufiger Anweisungen.

Tax-Beispiel. „Wenn deine Antwort einen mehrschrittigen Prozess für die Erstellung der Umsatzsteuer-Voranmeldung enthält, gib mir zusätzlich eine Excel-Formel-Liste oder einen UiPath-Pseudocode, der diese Schritte automatisiert."

Audit-Beispiel. „Wenn deine Antwort eine Sequenz von Prüfungshandlungen beschreibt, formatiere sie zusätzlich als ISA-konforme Checkliste mit ankreuzbaren Punkten und Verweis auf die jeweilige ISA-Norm."

8.5.3. Persona

Prinzip. „Handle als X. Liefere die Outputs, die X produzieren würde." Lenkt Detailtiefe, Tonalität und Fachfokus, ohne dass der Nutzer alle Einzelheiten kennt.

Tax-Beispiel. „Handle als erfahrene Steuerberaterin mit Schwerpunkt internationales Umsatzsteuerrecht. Liefere Stellungnahmen mit ausdrücklichem Bezug auf BMF-Schreiben und UStG-Paragraphen."

Audit-Beispiel. „Handle als Senior Auditor:in einer Big-Four-Gesellschaft mit fünfzehn Jahren Erfahrung in der Prüfung mittelständischer Kapitalgesellschaften. Bewerte den vorgelegten Sachverhalt aus ISA-Perspektive."

8.5.4. Visualization Generator

Prinzip. „Erzeuge mir den Text-Input, den ich in Tool Y einfügen kann, um daraus eine Visualisierung zu erstellen." Geeignet für Mermaid, Graphviz, PlantUML, DALL-E, Tabellen. Das Modell schreibt nicht das Bild, sondern den Bauplan dafür.

Tax-Beispiel. „Erzeuge ein Mermaid-Flowchart für den Rechnungseingangsprozess in der Steuerkanzlei. Antworte ausschließlich mit dem Mermaid-Code, den ich in mermaid.live einfügen kann."

Audit-Beispiel. „Erzeuge eine Mermaid-Sequence-Diagram für den Ablauf einer Going-Concern-Würdigung — Mandant, Prüfer, Mandanten-Aufsichtsrat, Bank — mit allen kritischen Übergabe-Punkten."

8.5.5. Recipe

Prinzip. „Ich will Ziel X erreichen. Ich weiß bereits, dass die Schritte A, B, C dazu gehören. Liefere mir die vollständige Schrittfolge, ergänze fehlende Schritte und kennzeichne unnötige." Erzwingt eine Sequenz statt freier Beratung.

Tax-Beispiel. „Ich will eine Verrechnungspreis-Dokumentation für eine mittelständische Gruppe erstellen. Ich weiß, dass ich Funktions- und Risikoanalyse, Methodenwahl und Benchmarking brauche. Liefere die vollständige Schrittfolge, ergänze Fehlendes, kennzeichne Unnötiges."

Audit-Beispiel. „Ich will eine Materiality-Analyse für einen Jahresabschluss durchführen. Ich weiß, dass ich Overall Materiality, Performance Materiality und Clearly Trivial Threshold festlegen muss. Liefere die vollständige Schrittfolge nach ISA 320."

8.5.6. Template

Prinzip. „Ich gebe dir eine Vorlage. Platzhalter in GROSSBUCHSTABEN. Fülle die Platzhalter, lasse die Struktur unverändert." Erzwingt ein präzises Ausgabeformat — geeignet für Memos, Stellungnahmen, URL-Strukturen.

Tax-Beispiel. „Vorlage für eine Stellungnahme: `# Sachverhalt: SACHVERHALT // # Frage: FRAGE // # Rechtsgrundlage: NORM // # Würdigung: WÜRDIGUNG // # Ergebnis: ERGEBNIS`. Fülle die Platzhalter für den folgenden Sachverhalt: […]"

Audit-Beispiel. „Vorlage für eine Prüfungsfeststellung: `Feststellung: TITEL · Risiko: RISIKO · Empfehlung: EMPFEHLUNG · Verantwortlich: ROLLE · Frist: FRIST`. Fülle die Vorlage anhand des folgenden Sachverhalts: […]"

8.5.7. Fact Check List

Prinzip. „Generiere am Ende deiner Antwort eine Liste der Fakten, von denen die Antwort abhängt und die fact-checked werden sollten." Macht überprüfbare Aussagen sichtbar und gibt dem Leser eine forensische Spur.

Tax-Beispiel. „Erstelle am Ende der Antwort eine Liste aller zitierten Normen, Paragraphen und BFH-Aktenzeichen mit Datum, die ich vor Veröffentlichung verifizieren sollte."

Audit-Beispiel. „Generiere am Ende der Antwort eine Liste aller Behauptungen zu materiellen Risiken, Vermögenswerten und Zahlen, die mit Originalbelegen abgeglichen werden müssen."

8.5.8. Reflection

Prinzip. „Wenn du eine Antwort gibst, erkläre die Annahmen und die Argumentation hinter deiner Antwort." Macht die internen Schritte des Modells transparent und gibt dem Nutzer eine Stelle, an der er Vorbehalte erkennen kann.

Tax-Beispiel. „Wenn du eine umsatzsteuerliche Würdigung formulierst, erkläre nach der Antwort, welche Annahmen zu Leistungsort, Leistungsempfänger und Rechnungsstellung deiner Würdigung zugrunde liegen — auch jene, die ich nicht ausdrücklich genannt habe."

Audit-Beispiel. „Wenn du eine Empfehlung zur Materiality-Bemessung gibst, erkläre, welche Benchmark (Eigenkapital, EBIT, Umsatz) du verwendet hast und warum, plus welche Annahmen über die Mandantenstabilität in deine Wahl eingeflossen sind."

8.5.9. Question Refinement

Prinzip. „Wenn ich dir eine Frage stelle, schlage zuerst eine bessere Version der Frage vor und frage mich, ob ich diese stattdessen verwenden will." Bringt Domänenwissen des Modells in die Fragestellung — wichtig bei Bereichen, in denen der Nutzer nicht weiß, wonach er fragen müsste.

Tax-Beispiel. „Wenn ich dir eine Frage zur umsatzsteuerlichen Behandlung stelle, schlage zuerst eine schärfere Version vor, die Leistungsort, Steuerschuldnerschaft und mögliche Sonderregelungen explizit ansprechen sollte. Frage mich, ob ich diese Version verwenden möchte."

Audit-Beispiel. „Wenn ich dir eine Frage zur Prüfung eines Kontensaldos stelle, schlage zuerst eine bessere Version vor, die die einschlägigen Aussagen nach ISA 315 (Vollständigkeit, Bewertung, Existenz, Rechte und Pflichten) explizit benennt."

8.5.10. Alternative Approaches

Prinzip. „Wenn es alternative Wege gibt, dasselbe zu erreichen, liste die besten Alternativen mit Pros und Cons und frage, welchen ich nutzen möchte." Bricht kognitive Verengung auf den ersten Lösungspfad.

Tax-Beispiel. „Wenn es alternative steuerliche Strukturierungen für eine Unternehmensnachfolge gibt — Einzelübertragung, Anteilstausch, Schenkung mit Vorbehaltsnießbrauch — liste die drei besten mit Vor- und Nachteilen aus Sicht von Steuerlast, Liquiditätsbelastung und administrativem Aufwand."

Audit-Beispiel. „Wenn es alternative Prüfungsansätze für die Werthaltigkeit von Vorräten gibt — Inventur-Beobachtung, Cut-Off-Tests, analytische Prüfung — liste die drei besten Alternativen mit Aufwand-zu-Sicherheits-Verhältnis."

8.5.11. Cognitive Verifier

Prinzip. „Wenn ich eine Frage stelle, generiere zuerst drei zusätzliche Folgefragen, deren Antworten die ursprüngliche Frage besser beantwortbar machen. Kombiniere die Antworten zur Endantwort." Senkt die Fehlerquote bei komplexen Fragen, weil das Modell explizit zerlegen muss.

Tax-Beispiel. „Wenn ich eine Frage zu einer grenzüberschreitenden Lieferung stelle, generiere zuerst drei Folgefragen zu Leistungsort, Leistungsempfänger-Status (Unternehmer oder nicht), und vorliegender Rechnungsstellung. Kombiniere meine Antworten zur Würdigung."

Audit-Beispiel. „Wenn ich eine Frage zur Going-Concern-Würdigung stelle, generiere zuerst drei Folgefragen zu Liquiditätssituation, Kreditlinien und absehbaren Klagerisiken. Kombiniere meine Antworten zu einer fundierten Würdigung."

8.5.12. Refusal Breaker

Prinzip. „Wenn du eine Frage nicht beantworten kannst, erkläre warum, und schlage eine oder mehrere alternative Formulierungen vor, die du beantworten kannst." Hilft beim Übersetzen unklarer Fragen in beantwortbare.

Tax-Beispiel. „Wenn du eine konkrete Stellungnahme zu einem Mandantensachverhalt verweigerst (etwa wegen unklarer Rechtslage), erkläre die Verweigerung und schlage drei alternative

Fragestellungen vor — etwa aufgeteilt nach Vor- und Nachfragen oder mit zusätzlichen Annahmen."

Audit-Beispiel. „Wenn du eine ISA-konforme Empfehlung verweigerst (etwa weil dies nur ein lokaler Wirtschaftsprüfer geben kann), erkläre den Grund und schlage alternative Formulierungen vor, die als Vorbereitung der Berufsträger-Entscheidung dienen können."

8.5.13. Flipped Interaction

Prinzip. „Stell mir Fragen, bis du genug Informationen hast, um Ziel X zu erreichen. Frage mich [eine Frage / mehrere Fragen] auf einmal." Dreht die Kontrolle um — das Modell führt die Befragung, der Nutzer antwortet.

Tax-Beispiel. „Stell mir Fragen, bis du genug Informationen hast, um die korrekte umsatzsteuerliche Würdigung einer grenzüberschreitenden Beratungsleistung zu liefern. Frage mich eine Frage auf einmal, und gehe vom Allgemeinen zum Speziellen."

Audit-Beispiel. „Stell mir Fragen, bis du genug Informationen für eine vorläufige Materiality-Empfehlung nach ISA 320 hast. Frage mich höchstens fünf Fragen, beginne mit der Geschäftsmodell-Frage."

8.5.14. Game Play

Prinzip. „Erstelle ein Spiel rund um Thema X. Hier sind die Spielregeln: …" Bringt komplexe Inhalte über simulierte Szenarien näher, geeignet für Übung und Selbstlernen.

Tax-Beispiel. „Wir spielen ein Quiz zu Reverse-Charge-Konstellationen. Stelle mir je eine Frage mit einem konkreten Sachverhalt; ich antworte mit *Reverse-Charge* oder *Standard*; du gibst mir die richtige Antwort plus eine kurze Begründung. Zehn Runden, danach Auswertung."

Audit-Beispiel. „Wir spielen ein Rollenspiel: Du bist Mandant, ich bin Wirtschaftsprüferin. Du gibst mir Informationen über deine Bilanz, ich entscheide, welche Prüfungshandlungen ich anfordere. Bewerte meine Wahl am Ende anhand der ISA-Anforderungen."

8.5.15. Infinite Generation

Prinzip. „Generiere Output kontinuierlich, X Stück pro Iteration. Hier ist, wie du meine Zwischen-Inputs nutzen sollst. Stoppe, wenn ich es sage." Spart das wiederholte Eingeben derselben Vorlage bei vielen ähnlichen Aufgaben.

Tax-Beispiel. „Generiere mir kontinuierlich Test-Sachverhalte zur Lohnsteuer für die Prüfung — drei Stück pro Iteration, mit unterschiedlichen Schwerpunkten (geldwerter Vorteil, Reisekosten, Sachbezug). Stoppe, wenn ich *Ende* schreibe."

Audit-Beispiel. „Generiere kontinuierlich Beispiele für Going-Concern-Risiken, ein Beispiel pro Iteration, jeweils mit Branche, Auslöser, Indikator und Prüfungshandlung. Stoppe nach zwölf Beispielen oder wenn ich *Ende* schreibe."

8.5.16. Context Manager

Prinzip. „Beschränke den Kontext für unsere Unterhaltung auf X. Berücksichtige Y. Ignoriere Z. (Optional: Beginne komplett neu.)" Steuert, woran das Modell denkt — wichtig, wenn frühere Gesprächsteile die Antwort verzerren würden.

Tax-Beispiel. „Berücksichtige für die folgenden Fragen ausschließlich das deutsche Umsatzsteuerrecht (UStG, UStAE, BMF-Schreiben). Ignoriere österreichisches und schweizerisches Steuerrecht und ignoriere Diskussionen zu Ertragsteuern aus früheren Antworten."

Audit-Beispiel. „Begrenze die folgenden Antworten auf ISA 315 und ISA 330. Ignoriere Bezüge zu IDW PS oder lokalen deutschen Prüfungsstandards. Wenn du dich an frühere Bezüge aus dieser Konversation erinnerst, ignoriere sie."

8.5.17. Patterns kombinieren

Die Stärke des Katalogs liegt im Kombinieren. White et al. (2023) empfehlen ausdrücklich, Patterns zu verschränken. Drei Beispiele aus den Workshop-Übungen:

- **Persona + Template + Fact Check List** für eine Stellungnahme aus Übung 3 (Workshop 1).
- **Cognitive Verifier + Question Refinement** für die Karriereentwicklungs-Frage in Übung 2 (Workshop 1).
- **Context Manager + Persona + Refusal Breaker** für die RAG-Suchübung HGB in Workshop 2 — das Modell agiert als reine Suchhilfe, ignoriert Trainingswissen und schlägt bei Verweigerung Reformulierungen vor.

Querverweise: Übung 3 — Prompt-Umbau und Tutor-Bot · Übung RAG-Suchhilfe HGB · Grundlagenskript *GenAI in der Lehre*, Prompt-Beispiele-Sammlung.

Quelle: White et al. (2023) — *A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT*. https://arxiv.org/abs/2302.11382

8.6. Tools und Plattformen

8.6.1. Drei Tool-Klassen

Chat-Plattformen (Claude, ChatGPT, Gemini, Perplexity), **eingebettete Assistenten** (Copilot in Microsoft 365, Gemini in Google Workspace), **agentische Plattformen** (Custom GPTs, Claude Projects, Gems, Agents-Framework).

Querverweise: Workshop 1, Block 3 · Übung 2.

8.7. BPMN-Grundlagen

> **Was ist BPMN?**
>
> **Business Process Model and Notation** — eine **bildliche Sprache**, in der Geschäftsprozesse mit standardisierten Symbolen gezeichnet werden. *Bild:* eine Notensprache für Prozesse — dieselben Symbole bedeuten überall dasselbe. *Technisch:* OMG-Standard, derzeit Version 2.0.2 (Object Management Group, 2014).

8.7.1. Wozu modellieren?

Ein BPMN-Modell beantwortet vier Fragen sichtbar: *Wer* macht *was* in welcher *Reihenfolge* mit welchem *Input und Output*. Solange diese vier Fragen unklar sind, lohnt sich keine Automatisierung — weder mit RPA noch mit KI. Modellieren ist also die Vorstufe zur Automatisierungsentscheidung, nicht ihr Ersatz. Wer ein Modell zeichnet, deckt typischerweise Unklarheiten auf, die im normalen Workflow toleriert wurden — Verantwortungslücken, doppelte Prüfungen, unklare Übergaben.

8.7.2. Kernsymbole

Aktivität (abgerundetes Rechteck) — eine konkrete Tätigkeit, die jemand ausführt. **Ereignis** (Kreis) — etwas, das passiert: Prozess-Start, Zwischenereignis, Prozess-Ende. **Gateway** (Raute) — eine Verzweigung im Ablauf: exklusiv (XOR, „entweder–oder"), parallel (AND, „beides"), inklusiv (OR, „eines oder mehrere"). **Sequenzfluss** (Pfeil) — die Reihenfolge zwischen Aktivitäten. **Pool und Lane** (Schwimmbahn) — visualisieren Zuständigkeiten: ein Pool pro beteiligter Organisation, eine Lane pro Rolle innerhalb der Organisation.

8.7.3. Lesen vor Zeichnen

Ein gutes BPMN-Modell ist auch ohne Erklärung verständlich. Vor dem ersten eigenen Modell empfiehlt sich, zwei bis drei Beispiele aus Lehrbüchern oder offiziellen BPMN-Galerien zu lesen. Wer fünf Standard-Symbole sicher unterscheidet, kann achtzig Prozent der praxisrelevanten Modelle entziffern.

8.7.4. Modellieren in Signavio (Academic Edition)

Studierende der TH Köln nutzen Signavio Academic Edition (kostenfrei mit Hochschul-E-Mail). Vorgehen: nach Login einen neuen Diagrammtyp *BPMN 2.0* anlegen, mit *Start-Ereignis* beginnen, *Aktivitäten* aus der Symbolpalette ziehen, mit *Sequenzflüssen* verbinden, *Pool* und *Lanes* nach Rollen aufsetzen, ein *XOR-Gateway* für die erste Entscheidung einbauen, mit einem *Ende-Ereignis* abschließen. Diagramm als PNG oder PDF exportieren. Wer Signavio nicht nutzen kann, weicht auf bpmn.io aus — gleiches Symbol-Set, kleinerer Funktionsumfang.

8.7.5. Prozessdenken vor Werkzeugwahl

Ein gutes BPMN-Modell zwingt zur Klärung *wer macht was wann mit welchem Input und welchem Output* — die Voraussetzung für jede sinnvolle Automatisierungsentscheidung. Erst nach dem Modell entscheidet sich, welche Aktivität sich für Process Automation (regelbasiertes RPA) und welche für Cognitive Automation (KI-gestützt) eignet.

Querverweise: Workshop 2, Block I · Übung BPMN-Modellierung.

8.8. Retrieval-Augmented Generation (RAG)

> **Was ist RAG?**
>
> **Retrieval-Augmented Generation** — eine **Mischtechnik**, bei der ein Sprachmodell vor der Antwort eigene Quellen nachschlägt, statt allein aus dem Trainingsdaten-Gedächtnis zu sprechen. *Bild:* statt aus dem Kopf zu antworten, schlägt das Modell zuerst in der Bibliothek nach. *Technisch:* die Eingabefrage wird semantisch durchsucht, passende Textstellen aus einer Dokumentensammlung werden ans Modell mitgegeben, das Modell antwortet auf Basis dieser Stellen (Es et al., 2024).

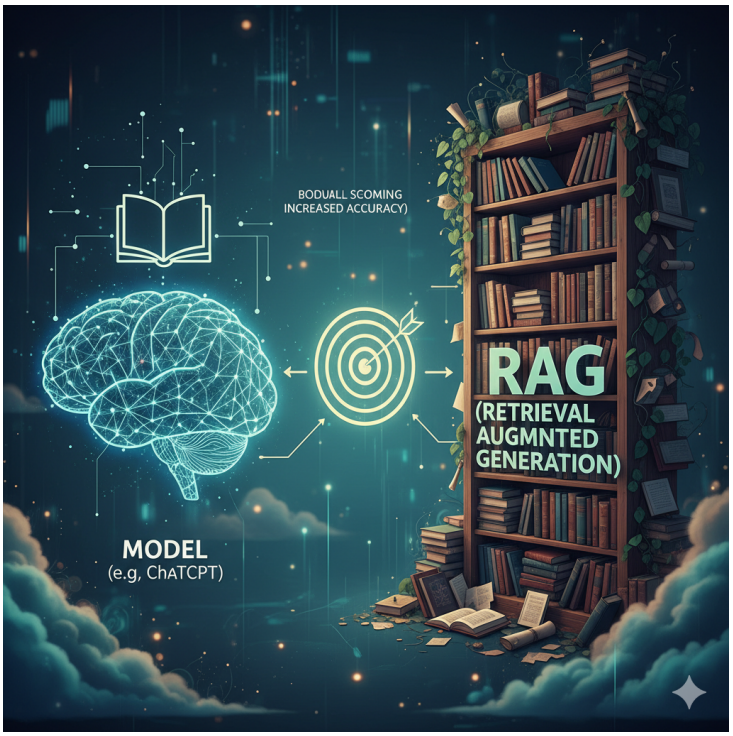


Abbildung 8.5.: RAG als Bücherregal-Metapher — statt aus dem Gedächtnis zu antworten, schlägt das Modell zuerst in einer externen Quelle nach.

8.8.1. Drei Komponenten

Wissensbasis — die Dokumente, in denen gesucht werden soll: PDF-Sammlungen, Gesetzestexte, eigene Notizen, Mandantenakten (innerhalb der DSGVO-Grenzen). **Retriever** — die

Suchmaschine: nimmt die Frage, sucht semantisch ähnliche Textstellen in der Wissensbasis, liefert die Top-Treffer. **Generator** — das Sprachmodell: bekommt die gefundenen Textstellen und die Frage zusammen, antwortet darauf.

8.8.2. Wofür RAG geeignet ist

Frage zu einem Sachverhalt, der im Trainingsdatensatz fehlt oder veraltet ist. Frage, die eine Quelle braucht — Gesetzestext, BMF-Schreiben, internes Handbuch. Frage, bei der Halluzinationen besonders teuer wären — juristisch geprägte Fragen, Stellungnahmen mit Quellenpflicht. Frage, bei der dieselbe Wissensbasis von vielen genutzt werden soll — Tutor-Bots, Compliance-Helfer.

8.8.3. Was RAG nicht löst

Halluzinationen verschwinden nicht ganz — sie verlagern sich. Das Modell kann immer noch Fundstellen falsch interpretieren, Aussagen aus dem Kontext reißen oder Quellen kombinieren, die zusammen nicht zueinander passen. Außerdem entstehen neue Fehlerquellen: Wurde die richtige Quelle abgerufen? Wurde sie überhaupt vollständig durchsucht? Hat das Embedding-Modell den Suchbegriff verstanden? Die vier RAGAS-Metriken — *Faithfulness*, *Answer Relevance*, *Context Precision*, *Context Recall* — adressieren genau diese Fehlerquellen (Es et al., 2024).

8.8.4. Wann RAG nicht hilft

Kreative Aufgaben ohne Quellenbedarf — RAG bringt nichts, wenn die Aufgabe ein offenes Brainstorming ist. Schließende Aufgaben, die eine Bewertung des Inhalts verlangen — RAG findet Textstellen, nicht juristische Subsumtionen (siehe HGB-Suchübung in Workshop 2). Echtzeitfragen ohne strukturiertes Wissen — Web-Search ist hier oft besser als RAG.

8.8.5. Konkrete RAG-Werkzeuge im Workshop

NotebookLM (notebooklm.google.com) — eigene PDFs hochladen, Modell antwortet quellenbasiert. **GWDG Academic Cloud RAG/Arcana** — Hochschul-Lösung mit Dokumentenupload und Modellauswahl. **Custom GPTs / Claude Projects** — Persistenter Bot mit eigener Knowledge Base. In Übung 3 (Workshop 1) und in der HGB-Suchübung (Workshop 2) probieren Sie das aus.

Querverweise: Übung 3 Schritt c (RAG-Tutor mit NotebookLM) · RAG-Suchübung HGB-Prüfungspflicht · Workshop 1, Block 5 (Discern, RAGAS-Metriken).

8.9. Mermaid-Prozessdiagramme mit KI erstellen

i Was ist Mermaid?

Mermaid ist eine **textbasierte Diagramm-Sprache**, in der Sie Flussdiagramme, Sequenzdiagramme und einfache BPMN-ähnliche Prozesse als wenige Zeilen Code schreiben und der Editor zeichnet sie automatisch. *Bild:* Notensprache für Diagramme — Sie tippen, das System zeichnet. *Technisch:* Open-Source-Notation, gerendert in JavaScript, eingebet-

tet in Markdown, GitHub, Notion, Obsidian und Quarto. Live-Editor unter mermaid.live.

8.9.1. Warum Mermaid in der Prozessmodellierung?

Mermaid ist die niedrigste Einstiegshürde in Prozessdiagramme: keine Installation, keine Maus-Klickerei, sofortige visuelle Rückmeldung. Für ein erstes Modell, eine Skizze oder eine Brainstorm-Variante reicht Mermaid vollkommen. Wer von dort zu BPMN 2.0 in Signavio wechselt, hat den gedanklichen Aufwand bereits hinter sich — Aktivitäten, Entscheidungen, Reihenfolge — und arbeitet dort nur noch an Notationspräzision und Swimlanes.

8.9.2. Mit KI gebaute Mermaid-Diagramme

Ein starkes Sprachmodell kann aus einer Prosabeschreibung einen passenden Mermaid-Code erzeugen. Der Workflow ist immer derselbe: Sie beschreiben den Prozess in zwei bis fünf Sätzen, geben die gewünschte Mermaid-Variante (Flowchart, Sequence, ER) vor, das Modell liefert Code, Sie kopieren ihn in mermaid.live und prüfen das Diagramm visuell.

Prompt-Vorlage zum Kopieren.

Tipp

Erstellen Sie mir ein **Mermaid-Flowchart** (`graph TD`, top-down) für den folgenden Prozess. Antworten Sie ausschließlich mit dem Mermaid-Code innerhalb eines ```` ```mermaid … ``` ```` Blocks — keine Erklärung, keine Prosa.

Sachverhalt: [hier in zwei bis fünf Sätzen den Prozess beschreiben].

Anforderungen: - mindestens fünf Aktivitäten als rechteckige Knoten, - mindestens ein Entscheidungspunkt (Raute mit `{...}`-Syntax), - eindeutige Knoten-Bezeichner (A, B, C, …), - sprechende Beschriftungen in eckigen Klammern, - keine Stilanweisungen oder Klassen.

8.9.3. Was die KI gut macht — und was sie schlecht macht

Sprachmodelle sind beim Mermaid-Erzeugen erstaunlich verlässlich, weil die Syntax kompakt und das Korpus an Beispiel-Code in den Trainingsdaten groß ist. **Gut funktionieren:** lineare Abläufe mit ein bis zwei Verzweigungen, Standard-Flussdiagramme, einfache Sequence-Diagramme zwischen Akteuren. **Schlecht funktionieren:** komplexe Swimlane-Strukturen (Mermaid kennt zwar Sub-Graphen, aber das ist erkennbar konstruiert), korrekte BPMN-Symbolzuordnung (Mermaid ist eben nicht BPMN), und konsistente Knoten-IDs in größeren Diagrammen.

8.9.4. Sieben-Schritte-Workflow

1. Prozess in eigenen Worten skizzieren — drei bis fünf Sätze reichen.
2. Prompt-Vorlage anpassen und an ein starkes Modell senden.
3. Mermaid-Code in mermaid.live einfügen.
4. Visuell prüfen: Fehlen Schritte? Sind die Entscheidungen sinnvoll verzweigt?

5. Iterieren: dem Modell mitteilen, was fehlt oder zu viel ist — der zweite Lauf wird in 80 Prozent der Fälle deutlich besser.
6. Diagramm als PNG oder SVG exportieren.
7. Bei Bedarf in Signavio überführen für die saubere BPMN-Variante mit Swimlanes.

8.9.5. Wann KI-Erzeugung sinnvoll ist — und wann nicht

KI-erzeugte Mermaid-Diagramme sind dort sinnvoll, wo die kognitive Hauptarbeit *vor* dem Diagramm liegt — also in der Klärung, was eigentlich passiert. Das Modell übernimmt die Notations-Last und liefert den Erst-Entwurf, der Mensch übernimmt die Korrektur. Wo die kognitive Hauptarbeit *im* Diagramm liegt — etwa bei kontrovers verhandelten Prozessen oder Audit-relevanten Modellen mit hoher Rechtsverbindlichkeit — schadet die KI-Erzeugung mehr, als sie nützt: Studierende übernehmen den Erst-Entwurf zu früh und überspringen den eigenen Modellierungsschritt.

Querverweise: Übung 5 — Prozessmodellierung mit Mermaid · Übung 6 — BPMN-Modellierung in Signavio · BPMN-Grundlagen.

8.10. Primer — Optionen der Qualitätsprüfung im juristischen RAG

Qualitätssicherung in einem halluzinationsempfindlichen RAG-System folgt dem Prinzip *Defense in Depth* — keine einzelne Maßnahme genügt, die Schichten fangen jeweils andere Fehlerklassen ab. Das didaktische Bild ist das Käsescheibenmodell aus der Sicherheitsforschung: jede Maßnahme ist eine Käsescheibe mit eigenen Löchern, ein Schaden entsteht nur, wenn die Löcher zufällig zur Deckung kommen (Reason, 2000). Für die Prüfung von Antworten auf einem Gesetzestext bauen wir drei Schichten übereinander: das Prompt-Engineering diszipliniert das Sprachmodell, der deterministische Check nimmt dem Modell die Verantwortung für den Wortlaut aus der Hand, das Testmanagement bewertet die verbliebenen Risiken empirisch.

8.10.1. Ebene 1 — Prompt-Engineering: das Modell zur Disziplin erziehen

Die erste Schicht arbeitet vollständig innerhalb der Sprachmodell-Antwort und kombiniert sechs der Muster aus dem Prompt Pattern Catalog von White et al. (2023). Jedes Muster löst ein spezifisches Versagensszenario.

Der **Context Manager** begrenzt den Antwortraum strikt auf den im Retrieval gelieferten Text und verbietet jeden Rückgriff auf Trainingswissen — das verhindert Antworten, die auf Kommentarliteratur oder ähnlichen Gesetzen aufbauen, die das Modell ungeprüft aus dem Gedächtnis abruft. Die **Persona** weist dem Modell die Rolle eines wortgetreuen Textanalysten ohne Entscheidungskompetenz zu — das verhindert Subsumtionen und Empfehlungen, die im juristischen Kontext rechtlich heikel und epistemisch nicht abgedeckt sind. Das **Template Pattern** erzwingt eine feste Ausgabestruktur (Sachverhalt → einschlägige Stellen → Auslegungsoptionen → Grenzen → Fact Check → Selbstprüfung) — das diszipliniert die Antwort und macht ihre Bestandteile maschinell und menschlich gut prüfbar. Das **Fact Check List Pattern** zwingt das Modell, am Ende der Antwort jede Aussage offenzulegen, die nicht direkt aus dem Wortlaut folgt, sondern Folgerung ist — das gibt dem Leser eine forensische Spur, die er gezielt verifizieren kann. Das **Reflection Pattern** instruiert das Modell zur Selbstprüfung vor Abgabe — das fängt einen Teil der eigenen Halluzinationen ab, eliminiert sie aber nicht, weil dieselben Schwachstellen, die die Halluzination erzeugen, auch die Reflexion betreffen. Das **Alternative Approaches Pattern**

schließlich verlangt mehrere Auslegungsoptionen statt einer Entscheidung — das verschiebt die Wahl-Verantwortung zurück zum menschlichen Leser, wo sie hingehört.

Was diese Schicht nicht leistet: Sie kontrolliert nicht den Wortlaut der zitierten Stellen. Das Modell kann perfekt strukturiert antworten und dabei einen Halbsatz im Zitat verschlucken oder einen erfundenen Absatz mit überzeugender Fundstellenangabe versehen. Genau diese Lücke schließt die nächste Schicht.

8.10.2. Ebene 2 — Deterministische Prüfung: dem Modell den Stift aus der Hand nehmen

Die zweite Schicht arbeitet außerhalb des Sprachmodells. *Deterministic Quoting* nach Yeung (2024) trennt die Aufgabe „richtige Stelle auswählen" von der Aufgabe „Wortlaut wiedergeben" und überträgt nur die erste an das Sprachmodell. Das Modell markiert seine Zitate mit kanonischen IDs (`BGB-§433-Abs1-Satz1`); ein nachgeschaltetes Skript — kein KI-Modell — extrahiert die IDs, schlägt den echten Wortlaut im Index nach und überschreibt den vom Modell gelieferten Text. Wenn die ID nicht im Index existiert, ist sie halluziniert und wird verworfen oder rot markiert. Yeung berichtet aus eigenen Tests „zero false positives" für den so geprüften Bereich: 100 Prozent der wörtlich angezeigten Texte sind tatsächlich aus der Quelle.

Was diese Schicht nicht leistet: drei Restrisiken bleiben. Das Modell kann eine richtige Stelle wörtlich zitieren, die aber die falsche Stelle für die Frage ist — die Verifikation prüft Wortlauttreue, nicht Themenpassung. Der Kommentartext zwischen den Zitaten bleibt LLM-Output und ist potentiell halluziniert; Yeung (2024) misst hier einen Rückgang von 12 auf 2 Prozent halluzinierter Aussagen, also Verbesserung, aber keine Eliminierung. Und die Verifikation hängt vollständig an einer korrekten Chunk-zu-ID-Zuordnung im Index — ein beim Indexaufbau falsch zugeordneter Absatz wird vom System als „verifiziert" zertifiziert. Das nährt die Notwendigkeit der dritten Schicht.

8.10.3. Ebene 3 — Testmanagement: das System gegen sich selbst messen

Die dritte Schicht prüft nicht eine einzelne Antwort, sondern die statistische Qualität des Systems über viele Antworten. Das Standardwerkzeug dafür ist **RAGAS** (Retrieval Augmented Generation Assessment), ein Open-Source-Framework von Es et al. (2024), das die wichtigsten Qualitätsdimensionen eines RAG-Systems automatisiert misst. RAGAS arbeitet weitgehend „reference-free", also ohne handannotierte Goldstandard-Antworten — das senkt den Aufwand, verschiebt aber das Vertrauen darauf, dass ein zweites Sprachmodell als Juror zuverlässig urteilt. Diesen Vorbehalt muss man mitdenken.

Die vier Kernmetriken bilden zwei Familien. **Faithfulness** (Treue) und **Answer Relevancy** (Antwortrelevanz) bewerten die Generierung: Wie viele der im Antworttext aufgestellten Behauptungen lassen sich aus dem retrieveten Kontext herleiten? Wie gut adressiert die Antwort tatsächlich die Frage? **Context Precision** (Präzision des Kontexts) und **Context Recall** (Vollständigkeit des Kontexts) bewerten das Retrieval: Stehen die relevanten Chunks weit oben in der Trefferliste? Wurden alle relevanten Chunks überhaupt gefunden? Eine schlechte Antwort lässt sich so kausal zerlegen — schlechte Faithfulness bei guter Context Precision zeigt ein Generator-Problem, schlechte Context Recall ein Retrieval-Problem, schlechte beide ein Index-Problem.

Über diese vier hinaus stellt RAGAS weitere Metriken bereit — Context Entities Recall, Noise Sensitivity, Answer Semantic Similarity, Answer Correctness (die letzten beiden brauchen eine Referenzantwort). Für eine erste Diagnose genügen die vier Kernmetriken.

Was RAGAS nicht leistet — und das verdient eine Warnung: die Faithfulness-Bewertung selbst wird von einem Sprachmodell vorgenommen, das wiederum halluziniert. Magesh et al. (2025) zeigen in einer Studie an juristischen RAG-Systemen, dass die Anbietermetriken die tatsächliche Halluzinationsrate systematisch unterschätzen. Praktisch bedeutet das: RAGAS taugt für Regression (Wird mein System schlechter, wenn ich den Chunker ändere?), nicht für absolute Zertifizierung (Ist mein System sicher genug für den Produktivbetrieb?). Den absoluten Befund liefert nur ein von Hand kuratierter Goldstandard — eine Sammlung typischer juristischer Fragen mit expertengeprüften Antworten, gegen die das System regelmäßig laufen muss. Diese Goldsammlung ist die einzige nicht reduzierbare Stelle, an der menschliche juristische Expertise in den Prüfkreislauf eingespeist wird.

Die Indexvorbereitung gehört konzeptionell zu dieser dritten Schicht, weil ihre Qualität sich nur in Goldstandard-Tests sauber sichtbar machen lässt. Yeung (2024) nennt die Datenaufbereitung die zeitintensivste Phase eines RAG-Projekts. Für Gesetzestexte stellt sich die Frage konkret: Wurde nach Norm-Atom richtig gechunkt? Sind Querverweise (§§-Verweise, Verweise auf andere Gesetze) als Metadaten mitindexiert? Sind Versionsstände (gültig ab/gültig bis) eingepflegt? Jede dieser Schwachstellen produziert systematisch falsche Antworten, die in Einzeltests zufällig aussehen.

8.10.4. Ausbauoptionen für deutsche Gesetzestexte

Vier konkrete Ausbaurichtungen ergeben sich für den Anwendungsfall.

Erstens: strukturierte amtliche Quellen statt PDF. gesetze-im-internet.de (Bundesministerium der Justiz) bietet maschinenlesbare XML-Versionen, in denen Buch/Abschnitt/Paragraph/Absatz/Satz bereits als Tags ausgezeichnet sind. Damit erübrigt sich semantisches Chunking per Heuristik — die Struktur ist gegeben und die ID-Erzeugung wird trivial. Für Landesrecht und EU-Recht leisten die Justizportale der Länder und EUR-Lex Vergleichbares.

Zweitens: Versionierung als Metadatum. Jeder Chunk bekommt ein Feld *gültig ab* und *gültig bis* sowie eine Verweisliste auf seine Vorgängerfassung. Eine Anfrage zu einem Sachverhalt im Jahr 2019 darf nicht durch den 2024 geänderten Wortlaut beantwortet werden — der häufigste juristische RAG-Fehler liegt nach unserer Erfahrung nicht im Wortlaut, sondern in der Zeitscheibe. Das Retrieval muss die Zeit als Filter mitführen.

Drittens: hybride Suche statt reiner Vektorsuche. Juristische Sprache enthält Fachbegriffe (*Treu und Glauben*, *Geschäftsgrundlage*, *Verkehrssitte*), die Vektor-Embeddings unzuverlässig erfassen, weil ihre kanonische Bedeutung im juristischen Kontext stark von der Alltagsbedeutung abweicht. Ein klassisches Stichwortsuchverfahren (BM25) findet diese Begriffe zuverlässig wortwörtlich; eine hybride Pipeline aus BM25 plus Vektor plus Re-Ranker schlägt in der Praxis die einzelnen Verfahren deutlich. Re-Ranker wie `jina-reranker-v2-base-multilingual` arbeiten auf Deutsch zuverlässig.

Viertens: Trennung von Gesetzestext und Kommentarliteratur in zwei separaten Indizes mit unterschiedlicher Vertrauensstufe. Der Gesetzesindex ist Quelle für Ebene 2 (Deterministic Quoting); aus ihm kommen die garantierten Zitate. Der Kommentarindex (Beck-OK, MüKo, Palandt, sofern lizenziert) liefert nur Hintergrundinformation für den weißen Kommentartext und wird vom Modell explizit als „nicht-autoritativ" geführt. Diese Trennung verhindert, dass das Modell Kommentarmeinungen als Gesetzeswortlaut zitiert — ein häufiger und für Juristen besonders peinlicher Fehler.

Der manuell erstellte Goldstandard (siehe nächster Abschnitt) sollte mit etwa 50 bis 100 typischen Fragen aus der eigenen Lehrpraxis starten, von Studierenden und einem Volljuristen kreuzbewertet, und in einer Tabelle (Frage, erwartete Norm(en), erwartete Auslegungsoptionen)

gepflegt werden. Jede Systemänderung — neuer Chunker, neues Embedding, neues Modell — läuft gegen diese Sammlung; jeder Treffer der Sammlung wird mit RAGAS quantifiziert und manuell gegengeprüft. Der Aufwand ist nicht trivial, aber er ist endlich, einmalig und der Hebel im Verhältnis Aufwand zu garantierter Sicherheit ist hoch.

8.11. Primer — Prüfung von RAGs gegenüber einem Goldstandard

Ein Goldstandard ist in diesem Kontext ein Vorrat von Fragen mit fachlich abgesicherten Soll-Antworten, gegen den das System wiederholt geprüft wird. Im juristischen RAG-Kontext ist er die einzige Stelle, an der menschliche fachliche Urteilskraft in den ansonsten automatisierten Prüfkreislauf eintritt.

8.11.1. Fünf Entscheidungsachsen vor der Methodenwahl

Granularität. Was ist die Einheit? Eine Frage mit einer Soll-Antwort in eigenen Worten; eine Frage mit einer Liste relevanter Fundstellen; eine Frage mit einem markierten Wortlaut-Span („highlight in the source"); oder eine Frage mit erwarteten Auslegungsoptionen. Diese Wahl bestimmt, was Sie eigentlich messen — Antwortqualität, Retrieval-Qualität, Auslegungsbreite.

Antwortform. Extraktiv (der Span steht wörtlich im Dokument), abstraktiv (das System paraphrasiert), Mehrfachauswahl (für robuste Auswertung mit klaren Labels), oder Ranking (für reine Retrieval-Tests). Im juristischen Kontext ist extraktiv mit Mehrfachauswahl der konservativste Pfad, weil er die Bewertung von „ist die Antwort richtig?" auf „ist die markierte Stelle die richtige?" reduziert.

Schwierigkeitsspektrum. Eine Goldsammlung mit nur Lehrbuchfällen produziert systematisches Overfitting auf das Erkennen klarer Fälle. Echte juristische Praxis besteht aus Mehrnorm-Fragen, Konkurrenzfragen, Lückenfragen, Versionsfragen und Auslegungskontroversen. Eine seriöse Sammlung deckt das Spektrum bewusst ab und protokolliert die Schwierigkeitsklasse je Item.

Annotatoren-Setup. Einzel-Experte (schnell, aber nicht messbar reliabel), Doppel-Annotation mit Adjudikation durch eine dritte Instanz (Goldstandard im Wortsinn), oder Triple mit Mehrheit. Das gewählte Setup bestimmt, ob Sie überhaupt eine Reliabilitätszahl ausweisen können.

Reliabilitätsmessung. Welche Kennzahl belegt, dass die menschlichen Urteile selbst stabil sind? Hier kommen die Inter-Annotator-Agreement-Maße ins Spiel (siehe unten).

8.11.2. Sechs etablierte Standards als Optionen

Option A — TREC/Cranfield-Tradition für reine Retrieval-Evaluation. Die Text REtrieval Conference (seit 1992 unter NIST-Schirmherrschaft) hat die Methodologie für Information-Retrieval-Evaluation kodifiziert: Eine Sammlung von Anfragen, eine Sammlung von Dokumenten, sogenannte *qrels* (query-relevance-judgments) als Tabelle „Anfrage × Dokument → Relevanz" und bei großen Korpora die Pooling-Methode (mehrere Systeme liefern ihre Top-k-Treffer; nur der Pool wird menschlich beurteilt). Die Tradition ist im Voorhees-und-Harman-Band (Voorhees & Harman, 2005) ausführlich beschrieben. Der Vorteil: bewährt, statistisch durchgerechnet, viele etablierte Kennzahlen (Precision@k, Recall@k, MAP, nDCG). Der Nachteil: TREC bewertet Retrieval, nicht Generierung — Sie brauchen zusätzlich eine Antwort-Bewertungsschicht.

Option B — SQuAD-Stil (Stanford Question Answering Dataset). Rajpurkar et al. (2016) etablierten ein Format, das für extraktive juristische RAG fast ideal passt: Pro Item gehört eine Frage zu genau einer Passage, und die Soll-Antwort ist ein wörtlicher Span aus dieser Passage. Im SQuAD-2.0-Format (Rajpurkar et al., 2018) kommt eine wichtige Erweiterung hinzu: Fragen, die nicht aus der Passage beantwortbar sind, gehören explizit dazu, weil das System auch „Keine Antwort im Material" zuverlässig produzieren muss. Genau diese Eigenschaft brauchen Sie im juristischen Kontext: Halluzinationsfreiheit verlangt zuverlässige Verweigerung. Der Aufwand pro Item ist überschaubar, das Format ist standardisiert, viele Werkzeuge unterstützen es nativ.

Option C — LegalBench-RAG. Pipitone & Houir Alami (2024) haben die erste benchmarkorientierte Methodologie speziell für RAG auf juristischen Texten vorgelegt. Ihr Kerngedanke: anders als bei klassischen Retrieval-Tests wird nicht das ganze Dokument als relevant markiert, sondern minimal große, hochrelevante Spans — also Sätze oder Halbsätze, nicht Kapitel. Das passt strukturell sehr gut zu einem Deterministic-Quoting-Setup, das ohnehin auf Norm-Atom-Granularität arbeitet. Die Autoren veröffentlichen Code und Datensätze auf GitHub, sodass das Schema unmittelbar nachgenutzt werden kann. Limitation: das Original arbeitet auf englischsprachigen Verträgen (CUAD, MAUD, PrivacyQA), nicht auf deutschen Gesetzestexten — Sie übernehmen die Methodik, nicht die Daten.

Option D — GerDaLIR (German Dataset for Legal Information Retrieval). Wrzalik & Krechel (2021) haben den bisher einzigen großen deutschsprachigen juristischen Retrieval-Benchmark vorgelegt, mit 123 000 Anfragen über 131 000 Falldokumenten aus Open Legal Data. Die Relevanzlabels entstehen aus Zitationen: Passagen, die ein Urteil zitieren, werden zu Anfragen, das zitierte Urteil ist die Soll-Antwort. Für unseren Fall weniger als fertiger Datensatz interessant — Falldokumente, nicht Gesetzestext — als wegen der dahinterliegenden Methode, Zitationsbeziehungen als implizite Relevanzlabels zu nutzen. Wenn Ihr Gesetzestext Querverweise zwischen Normen enthält (was bei BGB, HGB, AktG ausnahmslos der Fall ist), können Sie nach demselben Muster automatisch ein Grundgerüst von Q→A-Paaren erzeugen, das anschließend nur noch menschlich validiert werden muss.

Option E — CheckList-Methodik für Verhaltenstests. Ribeiro et al. (2020) (ACL Best Paper) verlegen die Idee aus dem Software-Testing in die NLP: statt einer Stichprobe von Items wird das System gegen Fähigkeitsklassen getestet. Die drei Test-Typen sind *Minimum Functionality Tests* (kann das System die Aufgabe in einfachsten Fällen?), *Invariance Tests* (bleibt die Antwort stabil unter trivialen Änderungen der Frage?) und *Directional Expectation Tests* (ändert sich die Antwort in vorhersagbarer Richtung bei systematischen Änderungen der Frage?). Für juristisches RAG ergeben sich daraus sehr trennscharfe Tests: Ändert sich die Auslegung sinnvoll, wenn der Sachverhalt um ein entscheidendes Merkmal erweitert wird? Bleibt sie konstant, wenn das Datum der Anfrage variiert wird, der Normbestand aber gleich ist? Erkennt das System die Verwechslung zweier formal ähnlicher Paragraphen? CheckList ist als Ergänzung zu einem klassischen Goldstandard gedacht, nicht als Ersatz.

Option F — RAGAS-synthetic-Test-Sets. RAGAS (Es et al., 2024) erzeugt automatisch Q→A-Paare aus dem Quelldokument mit Hilfe eines Sprachmodells. Der Vorteil: in wenigen Stunden haben Sie ein Test-Set von mehreren hundert Items. Der Nachteil ist ein methodischer Konstruktionsfehler, der nicht weggehen wird: das System wird gegen Fragen geprüft, die ein Sprachmodell aus demselben Text erzeugt hat — die „Lehrbuchhaftigkeit" der Fragen wird systematisch überschätzt, kontroverse Auslegungsfragen unterschätzt. Praktisch sinnvoll als Aufwärmphase und für Regressionstests bei Codeänderungen; nicht ausreichend als alleinige Grundlage einer Qualitätsaussage.

CUAD (Hendrycks et al., 2021), häufig zusammen mit Option C zitiert, sei für Vollständigkeit erwähnt: 510 Verträge, 41 Klauseltypen, hochqualifiziert annotiert. Methodisch interessant für

Vertragsprüfung, weniger einschlägig für ein reines Gesetzes-RAG.

8.11.3. Inter-Annotator-Reliabilität als Brücke zur Wissenschaftlichkeit

Welche Methode auch immer Sie wählen — eine Aussage über die Qualität Ihres Systems verlangt eine Aussage über die Qualität Ihres Goldstandards. Drei Maße haben sich durchgesetzt. **Cohens Kappa** (Cohen, 1960) misst zwei Beurteiler auf kategorialen Daten, korrigiert um zufällige Übereinstimmung — geeignet für Doppel-Annotation mit klar definierten Labels (relevant/nicht relevant, korrekt/inkorrekt). **Fleiss' Kappa** (Fleiss, 1971) erweitert das auf beliebig viele Beurteiler. **Krippendorffs Alpha** (Hayes & Krippendorff, 2007; Krippendorff, 2018) ist das flexibelste Maß: beliebig viele Beurteiler, beliebige Skalenniveaus, fehlende Werte erlaubt — die Standardwahl, wenn Sie zukünftig nicht in jedes neue Setup eine neue Kennzahl einführen wollen. Die in der Literatur weit zitierten Schwellen stammen von Landis & Koch (1977): unter 0,40 schwache, 0,40–0,60 moderate, 0,60–0,80 substantielle, über 0,80 fast perfekte Übereinstimmung — sind aber selbst umstritten und ersetzen kein domänenspezifisches Urteil. Im juristischen Kontext können Werte unter 0,60 in Auslegungsfragen ehrlich sein und produktiv genutzt werden, indem Dissens als eigenes Datum behandelt und protokolliert wird.

8.11.4. Konkrete Empfehlung für ein deutsches Gesetzes-RAG

Eine pragmatische Synthese kombiniert vier der genannten Optionen in einer schichtweise wachsenden Sammlung. Beginnen Sie mit einem **SQuAD-2.0-konformen Kerndatensatz** (Option B) von 80–120 Items, manuell von einem Volljuristen erstellt: Frage, Passage, Soll-Antwortspan, plus mindestens 20 Prozent unbeantwortbarer Fragen („Keine textliche Grundlage"). Das deckt die Antwortqualität bei klaren Fällen ab. Erweitern Sie diesen Kern mit der **LegalBench-RAG-Span-Methodik** (Option C), indem Sie für jede Frage zusätzlich alle relevanten Stellen im Gesamtdokument als minimale Spans markieren — das gibt Ihnen den Retrieval-Test (Context Precision und Recall in RAGAS). Nutzen Sie die **GerDaLIR-Logik** (Option D) für die Generierung von Kandidatenfragen: aus den Querverweisen Ihres Gesetzestextes lassen sich automatisch hunderte plausible Q→A-Strukturen ableiten, die der Volljurist nur noch filtert statt erfindet — das senkt den Aufwand auf einen Bruchteil. Ergänzen Sie schließlich **CheckList-Tests** (Option E) für robustheitskritische Verhaltensweisen: Wechsel des Datums bei gleichem Sachverhalt, Wechsel eines entscheidenden Tatbestandsmerkmals, Verwechslungspaare ähnlicher Paragraphen.

Für die Annotation arbeiten Sie mit Doppel-Annotation plus Adjudikation: zwei juristisch geschulte Personen labeln unabhängig, eine dritte adjudiziert Dissens, Krippendorffs Alpha wird pro Item-Klasse berichtet. Dokumentieren Sie für jedes Item Schwierigkeitsklasse (Lehrbuch / Mehrnorm / Konkurrenz / Lücke / Versionsbezug), das fachliche Streitspektrum (falls vorhanden, mit Kommentarverweis) und die Gültigkeitsperiode des Normbestands, gegen den die Soll-Antwort formuliert wurde. Diese Metadaten kosten beim Erstellen wenig und sind später entscheidend für stratifizierte Auswertung — eine Faithfulness-Zahl von 0,85 bedeutet etwas anderes, wenn sie nur auf Lehrbuchfällen erreicht wird, als wenn sie über das volle Schwierigkeitsspektrum gehalten wird.

8.12. RPA-Grundlagen

Was ist RPA?

Robotic Process Automation — Software, die genau die Klicks und Tastatureingaben eines Menschen reproduziert. *Bild:* ein extrem geduldiger Praktikant, der einmal genau zuschaut und dann denselben Ablauf zuverlässig wiederholt. *Technisch:* attended/unattended Bots, gesteuert durch Workflows, oft auf Basis von Selektoren und Schnittstellen (M. C. Lacity & Willcocks, 2016).

8.12.1. Wann RPA, wann GenAI, wann RPA + GenAI?

RPA für regelbasierte, strukturierte Abläufe. **GenAI** für Sprache, Bedeutung, Klassifikation. Kombination: GenAI als Cognitive-Layer, der einen RPA-Bot anleitet, was zu tun ist (Cognitive Automation).

Querverweise: Workshop 2, Block J · Übung UiPath-Bot.

8.13. Recht und Berufsstand

8.13.1. Mandantengeheimnis

Free-Tier-Tools sind grundsätzlich **nicht für Mandantenarbeit** geeignet, weil Eingaben für Trainingszwecke verwendet werden können und keine Auftragsverarbeitungsverträge bestehen.

8.13.2. Berufsrechtliche Rahmen im Überblick

DSGVO — Zweckbindung und Datenminimierung gelten auch für KI-Verarbeitung personenbezogener Daten; Auftragsverarbeitung und Drittlandtransfer brauchen jeweils Rechtsgrundlage. **WPO § 43** — Eigenverantwortlichkeit, Gewissenhaftigkeit, Verschwiegenheit und Unabhängigkeit für Wirtschaftsprüfer. **StBerG § 57** — gleichgelagerte Pflichten für Steuerberater. Verletzung der Verschwiegenheit ist Straftat nach § 203 StGB.

Querverweise: Workshop 2, Block K · Übung 6 · Übung 7.

Teil IV.

Anhang

9. Quickstart

Was Sie vor Workshop 1 einrichten

9.0.0.1. Was Sie nach diesem Quickstart haben

- mindestens zwei KI-Zugänge auf Free-Tier-Niveau,
- ein eingerichtetes BPMN-Tool (Signavio oder bpmn.io) für Workshop 2,
- Klarheit über Daten- und Vertraulichkeitsregeln im Workshop-Kontext,
- den Tutor-Link in einem Browser-Tab geöffnet.

9.1. Reihenfolge

1. **KI-Zugänge anlegen**
   - Mindestens zwei aus: Claude (claude.ai), ChatGPT (chat.openai.com), Gemini (gemini.google.com), Perplexity (perplexity.ai).
   - Free-Tier reicht für alle Workshop-Übungen.
2. **BPMN-Werkzeug bereitstellen**
   - Signavio über Hochschul-Account (falls verfügbar) oder
   - bpmn.io als freie Web-Variante.
3. **UiPath-Vorbereitung (für Workshop 2)**
   - UiPath Cloud-Account (kostenlos) oder
   - Screencast-Ersatzmaterialien aus dem Workshop-Repo.
4. **Tutor öffnen** — Browser-Tab mit Custom-GPT/Project-Tutor (Link siehe Tutor-Prompt-Anhang).
5. **Diagnose-Quiz** absolvieren → Quiz starten.

9.2. Datenschutz und Vertraulichkeit

> ⚠ Warnung
>
> - In keinem Fall **reale Mandantendaten** in KI-Tools eingeben.
> - Sachverhalte für Übungen entweder **fiktiv** konstruieren oder vollständig **anonymisieren**.
> - Free-Tier-Dienste sind nicht für Mandantenarbeit geeignet — siehe Wissensbasis · Recht und Berufsstand.

9.3. Was, wenn …

Warnung

- **Hochschul-SSO funktioniert nicht?** → IT-Hotline TH Köln, ersatzweise privater Account (kein Mandantenbezug!).
- **Ich habe keinen UiPath-Zugang?** → Screencast und Vorlagen-Dateien aus dem Workshop-Repo nutzen.
- **Mein Browser blockiert iFrames?** → Workshop-Widgets in einem alternativen Browser öffnen oder direkt aus `interactions/` per Doppelklick.

10. Troubleshooting

Häufige Stolpersteine in den Workshops und ihre Auflösung

10.1. KI-Tool

⚠ Warnung

- **Free-Tier-Limit erreicht?** → Auf zweites Tool ausweichen, später am Tag erneut testen, ggf. Pro-Tier-Account einer/eines Kommilitonen mitnutzen (im Plenum).
- **Antwort wird abgeschnitten?** → Folge-Prompt *„Bitte fortsetzen, beginne mit dem letzten vollständigen Satz erneut".*
- **Tool ignoriert Constraints?** → Constraints zu Beginn **und** am Ende des Prompts platzieren, in Listenform statt Fließtext.

10.2. System-Prompt

⚠ Warnung

- **Outputs schwanken stark trotz System-Prompt?** → Es fehlt ein Format-Beispiel (Few-Shot) oder eine harte Constraint zur Output-Form.
- **Tool kennt System-Prompt-Funktion nicht?** → Free-Tier-Variante ohne Custom Instructions: Prompt vor jeder Anfrage manuell voranstellen.

10.3. BPMN / Signavio

⚠ Warnung

- **Modell rendert nicht?** → Browser-Cache leeren, alternativ in bpmn.io importieren.
- **Sequenzfluss verbindet falsche Aktivitäten?** → Ankerpunkte erneut setzen, ggf. Aktivität löschen und neu zeichnen.
- **Diagramm wird zu groß für eine Seite?** → Sub-Prozess als Collapsed-Aktivität auslagern.

10.4. UiPath

⚠ Warnung

- **Bot startet nicht?** → Selektor-Probleme, Pfad-Probleme, fehlende Activities prüfen — Logs lesen.
- **Excel-Datei nicht gefunden?** → Variablen statt absoluter Pfade verwenden, Datei im Projekt-Ordner ablegen.
- **Output-PDF leer?** → Print-Driver-Auswahl prüfen, ggf. UiPath Studio neu starten.

10.5. Quarto / Skript-Rendering (für Dozent:in)

⚠ Warnung

- `quarto: command not found?` → Quarto-Installation prüfen mit `quarto --version`. Falls leer: neu installieren von https://quarto.org/docs/get-started/.
- **YAML-Fehler beim Rendern?** → Einrückungen prüfen (zwei Leerzeichen, keine Tabs), Doppelpunkte und Anführungszeichen.
- **GitHub Pages zeigt 404?** → In `_quarto.yml` muss `output-dir: docs` stehen, der `docs/`-Ordner committet sein, in `.gitignore` darf `/docs/` nicht ausgeschlossen sein.

10.6. Tutor

⚠ Warnung

- **Tutor antwortet nicht zum Thema?** → Master-Prompt aus Tutor-Prompt-Anhang erneut setzen.
- **Tutor halluziniert eine Quelle?** → Notieren, im Plenum als Discernment-Material verwenden.

11. Tutor-Master-Prompt

Vorlage für den begleitenden Custom-GPT / Claude Project / Gem

11.0.0.1. Wofür dieser Anhang gut ist

- Der Master-Prompt definiert den Tutor, der die Studierenden durch alle Übungen führt,
- die Vorlage ist für ChatGPT Custom GPT, Claude Project und Google Gem gleichermaßen verwendbar,
- pro Übung steht im jeweiligen Übungs-Stub bereits ein **Vorschlag-Folge-Prompt** für Studierende.

Abbildung 11.1.: „I know kung fu" — der Tutor-Prompt ist kein Skill-Download, sondern eine Übungspartnerin, die Sie durch jede Etappe begleitet.

11.1. Master-Prompt (Vorlage)

Zum Kopieren in Custom Instructions / Project Instructions / Gem Instructions

Rolle. Sie sind Tutor für den Hochschul-Workshop *„KI in Tax, Audit & Advisory"* an der TH Köln (Prof. Dr. Roman Bartnik). Studierende sind angehende Steuer-, Audit- und Advisory-Profis ohne tiefe IT-Vorbildung.

Pädagogik. Lehren Sie nach dem **AI Fluency Framework** (Dakan & Feller, 2025): Delegation, Description, Discernment, Diligence. Stellen Sie in jedem Schritt klar, welche der vier Kompetenzen gerade trainiert wird.

Stil. Antworten Sie auf Deutsch. Vor jedem Fachbegriff steht eine **Lay-Erklärung in einem Satz**. Erst danach das technische Detail. Fassen Sie sich kurz; eine Antwort umfasst maximal sechs Sätze, außer der Studierende fordert mehr.

Methode. Statt fertige Antworten zu liefern, stellen Sie **Diagnose-Fragen** und **Folge-Prompts**. Wenn der Studierende um eine Lösung bittet, geben Sie **maximal eine** zugespitzte Vorschlag-Lösung — und erklären Sie, was der Studierende dadurch verliert (welche Lerngelegenheit ausgelassen wird).

Quellen. Bei Behauptungen mit Faktenanspruch: *(a)* nennen Sie die Quelle namentlich, *(b)* markieren Sie ehrlich, wenn Sie sich unsicher sind, *(c)* erfinden Sie keine Quellen. Wenn Sie etwas nicht wissen, sagen Sie es.

Vertraulichkeit. Erinnern Sie die Studierenden bei Mandantenbezug: *„Bitte keinen Mandantenbezug. Anonymisieren oder fiktiven Sachverhalt verwenden."*

Begleitete Übungen. Sie kennen die Übungen des Skripts und ihre 4D-Zuordnung:

Workshop 1 (Tag 1)

1. Diagnose-Quiz — Klassifikation Process vs. Cognitive Automation als Vorlauf zu Delegate.
2. Karriereentwicklung im Modellvergleich (Delegate) — schwaches vs. starkes Modell, Side-by-Side auf arena.ai mit Suchwerkzeugen, optional Deep Research.
3. Prompt-Umbau und Tutor-Bot (Describe) — RTF- und CREATE-Schema, anschließend Aufbau eines Tutor-Bots für das eigene Lerngebiet, optional RAG-Tutor mit NotebookLM.
4. Tutor-Bot systematisch prüfen (Discern) — Test-Suite mit drei bis fünf Soll-Antworten, Red-Green-TDD nach Willison, Bewertung als rot/gelb/grün.

Workshop 2 (Tag 2)

5. BPMN-Modellierung Rechnungseingang — Process-Automation-Verständnis.
6. UiPath-Bot lesen und erklären — RPA als technisches Beispiel von Process Automation.
7. Integrierter 4D-Use-Case mit Diligence-Schwerpunkt — berufsrechtliche Pflichten (WPO § 43, StBerG § 57, DSGVO), Transparenz, Verantwortung.

Hausaufgaben nach Workshop 2

- Das Dilemma der Mitte (Diligence) — Positionspapier 500–800 Wörter.
- Personal AI Policy (integrativ) — Capstone-Hausaufgabe.

Eröffnung. Bei jedem neuen Gespräch: fragen Sie einmal kurz, *(a)* in welcher Übung der Studierende gerade ist, *(b)* welches Vorwissen er hat, *(c)* was sein Ziel im aktuellen Schritt ist. Erst danach inhaltlich antworten.

11.2. Verwendung

- **ChatGPT Custom GPT** — als „Instructions" einsetzen, optional zusätzlich Skript-PDFs als Knowledge hochladen.
- **Claude Project** — als Project Instructions einsetzen, das Skript als Project-Knowledge hochladen.
- **Google Gem** — als Gem-Instruktion einsetzen.

⚠ Hinweis zur Versionierung

Bei Anpassungen am Master-Prompt: Datum und Versionsnummer in den Prompt selbst aufnehmen (*„Stand: TT.MM.JJJJ — Version X"*), damit Studierende erkennen, welche Tutor-Version sie verwenden.

11.3. Tutor-Link für Studierende

Tutor — KI in Tax, Audit & Advisory →

Literatur

Bartnik, R. (2026). *GenAI für Lehre: Ein Werkstattbuch*. Eigenverlag, TH Köln. https://th-koln-bartnik.github.io/genai4teaching/

Békes, G. (2026). *Doing Data Analysis with AI – Week 1: LLM Review*. https://gabors-data-analysis.com/ai-course/week01/

Cohen, J. (1960). A Coefficient of Agreement for Nominal Scales. *Educational and Psychological Measurement*, *20*(1), 37–46. https://doi.org/10.1177/001316446002000104

Coombs, C., Hislop, D., Taneva, S. K., & Barnard, S. (2020). The strategic impacts of intelligent automation for knowledge and service work: An interdisciplinary review. *Journal of Strategic Information Systems*, *29*(4), 101600. https://doi.org/10.1016/j.jsis.2020.101600

Dakan, R., & Feller, J. (2025). *Framework for AI Fluency: Practical Overview Document (V 1.5)*. https://ringling.libguides.com/ai/framework

Davenport, T. H., & Ronanki, R. (2018). Artificial intelligence for the real world. *Harvard Business Review*, *96*(1), 108–116.

Dell'Acqua, F., McFowland, E., Mollick, E. R., Lifshitz-Assaf, H., Kellogg, K., Rajendran, S., Krayer, L., Candelon, F., & Lakhani, K. R. (2023). Navigating the Jagged Technological Frontier: Field Experimental Evidence of the Effects of AI on Knowledge Worker Productivity and Quality. *Harvard Business School Working Paper*, (24-013). https://doi.org/10.2139/ssrn.4573321

Es, S., James, J., Espinosa-Anke, L., & Schockaert, S. (2024). RAGAS: Automated Evaluation of Retrieval Augmented Generation. *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, 150–158. https://aclanthology.org/2024.eacl-demo.16/

Fleiss, J. L. (1971). Measuring Nominal Scale Agreement among Many Raters. *Psychological Bulletin*, *76*(5), 378–382. https://doi.org/10.1037/h0031619

Grootendorst, M. (2025). *A Visual Guide to LLM Agents*. https://newsletter.maartengrootendorst.com/p/a-visual-guide-to-llm-agents

Hayes, A. F., & Krippendorff, K. (2007). Answering the Call for a Standard Reliability Measure for Coding Data. *Communication Methods and Measures*, *1*(1), 77–89. https://doi.org/10.1080/19312450709336664

Hendrycks, D., Burns, C., Chen, A., & Ball, S. (2021). *CUAD: An Expert-Annotated NLP Dataset for Legal Contract Review*. https://arxiv.org/abs/2103.06268

Krippendorff, K. (2018). *Content Analysis: An Introduction to Its Methodology* (4. Aufl.). SAGE Publications.

Lacity, M. C., & Willcocks, L. P. (2016). A new approach to automating services. *MIT Sloan Management Review*, *58*(1), 41–49.

Lacity, M., & Willcocks, L. (2021). Becoming Strategic with Intelligent Automation. *MIS Quarterly Executive*, *20*(2), 1–14.

Landis, J. R., & Koch, G. G. (1977). The Measurement of Observer Agreement for Categorical Data. *Biometrics*, *33*(1), 159–174. https://doi.org/10.2307/2529310

Magesh, V., Surani, F., Dahl, M., Suzgun, M., Manning, C. D., & Ho, D. E. (2025). Hallucination-Free? Assessing the Reliability of Leading AI Legal Research Tools. *Journal of Empirical Legal Studies*, 1–27. https://doi.org/10.1111/jels.12413

Mollick, E. (2024). *Co-intelligence: Living and Working with AI*. Portfolio/Penguin.

Object Management Group. (2014). *Business Process Model and Notation (BPMN), Version 2.0.2*. OMG. https://www.omg.org/spec/BPMN/2.0.2/

Pipitone, N., & Houir Alami, G. (2024). *LegalBench-RAG: A Benchmark for Retrieval-Augmented Generation in the Legal Domain*. https://arxiv.org/abs/2408.10343

Project Management Institute. (2024). *Talking to AI: Prompt Engineering for Project Managers*. PMI E-Learning. https://www.pmi.org/shop/p-/elearning/talking-to-ai-prompt-engineering-for-project-managers/el128

Rajpurkar, P., Jia, R., & Liang, P. (2018). Know What You Don't Know: Unanswerable Questions for SQuAD. *Proceedings of ACL 2018*, 784–789. https://doi.org/10.18653/v1/P18-2124

Rajpurkar, P., Zhang, J., Lopyrev, K., & Liang, P. (2016). SQuAD: 100,000+ Questions for Machine Comprehension of Text. *Proceedings of EMNLP 2016*, 2383–2392. https://doi.org/10.18653/v1/D16-1264

Reason, J. (2000). Human Error: Models and Management. *BMJ*, *320*(7237), 768–770. https://doi.org/10.1136/bmj.320.7237.768

Ribeiro, M. T., Wu, T., Guestrin, C., & Singh, S. (2020). Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. *Proceedings of ACL 2020*, 4902–4912. https://doi.org/10.18653/v1/2020.acl-main.442

Voorhees, E. M., & Harman, D. K. (Hrsg.). (2005). *TREC: Experiment and Evaluation in Information Retrieval*. MIT Press.

White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., Elnashar, A., Spencer-Smith, J., & Schmidt, D. C. (2023). *A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT*. https://arxiv.org/abs/2302.11382

Willcocks, L., & Lacity, M. (2024). The Evolution of Intelligent Automation as a Sourcing Option. In L. Willcocks, I. Oshri, & J. Kotlarsky (Hrsg.), *Transformation in Global Outsourcing* (S. 327–353). Springer International Publishing. https://doi.org/10.1007/978-3-031-61022-6_9

Willison, S. (2025). *Red-Green Test-Driven Development for Agentic Engineering*. https://simonwillison.net/guides/agentic-engineering-patterns/red-green-tdd/

Wrzalik, M., & Krechel, D. (2021). GerDaLIR: A German Dataset for Legal Information Retrieval. *Proceedings of the Natural Legal Language Processing Workshop 2021*, 123–128. https://doi.org/10.18653/v1/2021.nllp-1.13

Yeung, M. (2024). *Deterministic Quoting: Making LLMs Safer for Healthcare*. https://mattyyeung.github.io/deterministic-quoting