

KI-Expertenrat für Dokumente

Mehr Agenten – bessere Antworten?

Projektarbeit im Master-Studiengang Praktische Informatik · PIM-PA · Wintersemester 2026/27

Wie können mehrere spezialisierte KI-Agenten gemeinsam zuverlässigere und nachvollziehbarere Antworten zu Dokumenten erzeugen als ein einzelner Agent?

Ausgangspunkt

Sprachmodelle beantworten Fragen zu Dokumenten beeindruckend flüssig, können aber unvollständige, falsche oder nur scheinbar belegte Aussagen liefern. Ein virtueller Expertenrat soll helfen: Spezialisierte KI-Agenten entwickeln unabhängig Vorschläge, hinterfragen sich und prüfen sie am Dokument. Doch mehr Agenten bedeuten nicht automatisch mehr Qualität – sie können ebenso Kosten, Laufzeit und Fehler verstärken.

Die Projektvision

Nutzer:innen importieren Dokumente und fragen das System. Es findet relevante Passagen und übergibt sie einem konfigurierbaren Expertenrat. Dessen Agenten analysieren, kritisieren und verdichten ihre Ergebnisse zu einer verständlichen Antwort mit überprüfbaren Fundstellen. Fehlt Evidenz, macht das System seine Grenzen transparent.

Vom Dokument zur belegten Antwort

Dokumente → Retrieval → Analysen → Kritik und Quellenprüfung → Synthese

Ihr Projektauftrag

Sie entwickeln im Team einen lauffähigen Prototyp mit textbasierter Benutzeroberfläche und untersuchen empirisch, wann der Expertenrat gegenüber einem einfachen Single-Agent-RAG-System messbaren Mehrwert bietet. So verbinden Sie **Engineering** – ein modulares, beobachtbares und qualitätsgesichertes System – mit **Evaluation**: Varianten werden reproduzierbar verglichen und evidenzbasiert beurteilt.

Was am Ende entstanden sein soll

- Textual-TUI für Dokumente, Fragen und Antworten,
- Single-Agent-RAG als belastbare Baseline,
- konfigurierbarer Expertenrat mit festem Ablauf,
- nachvollziehbare Darstellung von Quellen und Evidenz,
- Evaluationsset und Vergleich von Qualität, Laufzeit und Kosten,
- automatisierte Tests und dokumentierte Architekturentscheidungen,
- Continuous Integration (CI).

Die spannenden Fragen

Die wesentliche Arbeit liegt in begründeten Designentscheidungen:

- Wie findet das System die entscheidenden Dokumentstellen?
- Welche Rollen und Protokolle schaffen echte Perspektivenvielfalt?
- Wie prüft das System Belege – und wann verweigert es eine Antwort?
- Wie werden Qualität, Laufzeit und Kosten fair mit der Baseline verglichen?

Technischer Rahmen und Gestaltungsfreiheit

Vorgegeben: Python, Textual-TUI, automatisierte Tests, CI und evaluierbare Baseline.

Voraussichtliche Werkzeuge: LangGraph, ausgewählte LangChain-Komponenten, Pydantic, pytest und Docker.

Ihre Entscheidungen: Retrieval, Agentenrollen, Beratungs- und Konsensverfahren, Datenhaltung und Qualitätsmetriken. Wesentliche Entscheidungen dokumentieren Sie als Architecture Decision Records (ADRs).

Arbeitsweise: Qualität von Anfang an

Sie entwickeln iterativ mit Kanban. Ein frühes Walking Skeleton bildet den Weg vom Dokument bis zur ersten belegten Antwort ab; anschließend folgen kleine vertikale Slices. Kurze Integrationszyklen, Pair oder Mob Programming und regelmäßige Reviews unterstützen das Team. Architektur, Testbarkeit, Beobachtbarkeit und Evaluation werden von Beginn an eingebaut. Zusätzliche Komplexität ist kein Fortschritt an sich: Entscheidend ist, ihren Nutzen gegenüber der Baseline nachvollziehbar zu untersuchen. Auch ein begründetes Verwerfen schafft Erkenntnis.

Was Sie dabei lernen und gestalten

Sie lernen, ein KI-System modular zu entwerfen, probabilistische Komponenten testbar zu machen und Entscheidungen evidenzbasiert zu treffen. Praktische Erfahrung sammeln Sie mit RAG, KI-Agenten, Workflow-Orchestrierung, Softwarearchitektur, automatisierter Evaluation, CI und agiler Teamarbeit.

Passt das Projekt zu Ihnen?

Das Projekt passt zu Ihnen, wenn Sie generative KI nicht nur verwenden, sondern kritisch hinterfragen möchten. Gefragt sind Freude an Softwareentwicklung im Team, Interesse an Architektur und Qualität sowie experimentelles, eigenverantwortliches Arbeiten. RAG-Vorkenntnisse sind hilfreich, aber nicht erforderlich.

Wichtig: Dies ist kein reines Prompt-Engineering-Projekt. Im Mittelpunkt stehen Software Engineering mit Built-in Quality sowie die kritische Evaluation des tatsächlichen Mehrwerts.

Prof. Dr.-Ing. Martin Burger

Software Engineering und Software Quality Assurance

martin.burger@htwsaar.de · sesqa.de

**ingenieur
wissenschaften
htw saar**