

Session 2.8: Danke! Nun geht's ans Projekt

Projekt: GenAI/Conversational AI Formular-Ausfüllassistent PoC (Proof of Concept)

Beschreibung

- Oft gibt es Formulare oder Fragebögen, die sehr spezifische Fragen beinhalten und mit denen sich User:innen überfordert fühlen
- Gerne hätten sie jemanden, um näher Fragen stellen zu können oder Vorschläge/Beispiele geben zu können
- Ein GenAI-Assistent hat das Potenzial, hier eine automatisierte beliebig vervielfältigbare Lösung darzustellen
- Conversational AI ist einer der Use Cases mit meistem Potenzial
- PoC = Minimal; zeigen, dass etwas geht oder nicht geht oder vielleicht gehen könnte

Umsetzung

- alles in ein Jupyter Notebook und während der Präsentation einzelne Schritte zeigen, wie diese bei so einem potenziellen Ausfüllassistenten aussehen könnte
- so wie wir es im Unterricht gemacht haben, nur spezieller bezogen auf euren UseCase
- keine fertige Anwendung, eher erforschen, wie etwas möglich sein könnte, warum, und warum nicht
- da wir mit kleinen Models lokal arbeiten, begrenzt in der Ausreizung der Modelle - daher lieber einzelne Schritte anstatt komplette Anwendung
- könnte man bei der Befüllung eines Feldes ein RAG-Tool nutzen?
- könnte ein Nutzer sich Beispiele geben lassen mit Hilfe eines RAG-Tools?

Use Case Beispiele

- Gerätesupport-Assistent
- Mitarbeiter-Richtlinien-Assistent
- Kontoeröffnungs-Assistent
- Versicherungsberatungs-Assistent
- Reiseplanungs-Assistent

- Bauantrags-Assistent
- Studienbewerbungs-Assistent
- Fördermittel-Antragsassistent
- Datenschutz-Compliance-Assistent
- Steuererklärungs-Assistent
- Patent-Anmeldeassistent
- Lebensmittel-Beratungsassistent
- ...

Mögliche Tool-Vorschläge

- RAG
- Wetter-API
- Währungsrechner-API
- Internetsuche
- Flug-/Zugverbindungs-API
- Kalendermanagement
- Openstreetmap
- Prozentrechner
- OpenFoodFacts (Lebensmittelinformationen)
- eigens kreierte einfache weitere Tools

Punkteverteilung Projektarbeit: Lieferobjekte

- 08P | 20 Seiten PDF (ca. 12k Wörter) zu einem Thema oder mehreren Themen einer Kategorie
- 10P | Document Processing - diese 20 Seiten PDF geparkt und gehunkelt
- 12P | Vektorisierung und Storage - diese 20 Seiten PDF embedded und indexiert in einer lokalen VektorDB
- 12P | Agent-Tool für RAG - Retrieval aus der Vektor-DB durch Embedding der Query mit Embedding-Model (from agents import FileSearchTool, file_search_tool = FileSearchTool)
- 08P | Form: Pydantic Model mit 10 Feldern, Typisierung und Description
- 20P | Umsetzung von 4 Tools (eines davon bereits RAG) mit Agenten, die zur Befüllung eines oder mehrerer der Felder beiträgt
- 15P | Anwendungsfunktionalität: alle Zellen und Beispiele lassen sich starten
- 15P | mind. 3 erfolgreiche Fälle, mind. 3 weniger erfolgreiche Fälle - genug Erfahrungen teilen
- GESAMT 100P -> macht 50% der Gesamtnote aus

Präsentation

- Infos

- 10 Minuten pro Gruppe
- Ihr könnt entscheiden, wer präsentiert - Punkte bekommt die ganze Gruppe
- Live-Durchführung durch Jupyter Notebook
- Abgabe: Notebook, 20-seitiges-PDF
- Punkteverteilung
 - 10P | Einhaltung 10 Minuten +/- 2 Minuten
 - 55P | Soll eine Live-Demo enthalten - Walkthrough durch Notebook
 - 10P | Soll ein Diagramm enthalten (Sequence, Prozess-Flow, ...)
 - 25P | Benützte Komponenten und Erfahrungen bei allen Schritten in der RAG-Pipeline
 - [Parsing -> Chunking -> Embedding -> Indexing -> Retrieval (Abruf) -> Augmented (erweiterte) Generation]
 - Welches Embedding-Model (welche Dimensionen), welches Generation-Model? Verschiedene ausprobiert?
 - Wurden vielleicht auch Agent Patterns angewendet?
 - GESAMT 100P -> macht 20% der Gesamtnote aus

Vorgangsweise

1. Entscheidung für Use Case und in Forum eintragen

1A) Einen dieser Use Cases für Gruppe aussuchen oder selbst einen erfinden

UC1. Gerätesupport-Assistent

- Felder
 - Gerätetyp (str)
 - Modellnummer (str)
 - Betriebssystemversion (str)
 - Fehlercode (Optional[str])
 - Fehler tritt regelmäßig auf (bool)
 - Akkulaufzeit in Stunden (Optional[int])
 - Letzter erfolgreicher Betrieb (Datum als str)
 - Gerät bereits neu gestartet (bool)
 - Beschreibung des Fehlverhaltens (str)
 - Garantie noch gültig (bool)
- Documents

- ca. 20 Seiten mit 15 verschiedenen Produkten und Produktdetails (z.B. 15 Drucker)
- Mögliche Tools
 - RAG für Abruf Gerätetyp, Modellnummer, Betriebssystemversion, ursprüngliche Akkulaufzeit für Vergleich
 - Akkulaufzeit-Unterschied-Rechner
 - Garantiezeitrechner
- Mögliche User:innen-Nachfrage-Prompts
 - "Was für ein Gerätetyp, Modellnummer und Betriebssystemversion hat Drucker xy?" -> ruft RAG-Tool auf
 - "Welche Fehlercodes gibt es für Drucker xy?" -> ruft RAG-Tool auf
 - "Was ist die ursprüngliche Akkulaufzeit für Drucker xy?" -> ruft RAG-Tool und Akkulaufzeit-Unterschied-Rechner auf
 - "Kannst du mir meine Beschreibung des Fehlverhaltens besser ausdrücken "lässt sich nicht mehr aufdrehen"? -> kein Tool-Aufruf
 - "Ich habe meinen Drucker xy am 12.03.2022 gekauft. Ist die Garantie noch gültig?" -> ruft Garantiezeitrechner-Tool auf

UC2. Mitarbeiter-Richtlinien-Assistent

- Felder
 - Beschäftigungsart (str, z.B. Vollzeit, Teilzeit)
 - Betriebszugehörigkeit in Monaten (int)
 - Anwendungsfall (str, z.B. Urlaub, Dienstreise)
 - Kenntnisnahme Richtlinien erfolgt (bool)
 - Datum letzter Schulung (Optional[str])
 - Verstoß gegen Richtlinie (bool)
 - Beschreibung des Vorfalls (Optional[str])
 - Führungskraft informiert (bool)
 - Betroffene Abteilung (str)
 - Zusätzliche Unterstützung benötigt (bool)
- Documents
 - Ca. 20 Seiten Mitarbeiter-Richtlinien, Urlaubsregeln, Homeoffice-Regelungen, Compliance-Regeln.
- Mögliche Tools
 - RAG für Abruf spezifischer Richtlinieninhalte
 - Beschäftigungszeit-Rechner (Umwandlung Monate zu Jahren)
 - Letztes-Schulungsdatum-Checker
- Mögliche User:innen-Nachfrage-Prompts
 - „Welche Urlaubsregelungen gelten für Teilzeit-Mitarbeiter?“ -> RAG

- „Ab wann muss ein Richtlinienverstoß gemeldet werden?“ -> RAG
- „Wie lang ist die Betriebszugehörigkeit in Jahren, wenn ich seit 37 Monaten hier bin?“ -> Beschäftigungszeit-Rechner
- „Wann fand zuletzt eine verpflichtende Datenschutz-Schulung statt?“ -> Letztes-Schulungsdatum-Checker
- „Hilf mir bitte, meine Beschreibung des Vorfalls klarer auszudrücken: ‚Kollege machte etwas falsch‘.“ -> Kein Tool-Aufruf

UC3. Kontoeröffnungs-Assistent

- Felder
 - Kontotyp (str)
 - Vorname Kontoinhaber (str)
 - Nachname Kontoinhaber (str)
 - Geburtsdatum (str)
 - Wohnsitzland (str)
 - Steuerpflicht in anderem Land (bool)
 - Monatliches Einkommen in € (int)
 - Politisch exponierte Person (bool)
 - Quelle der Einzahlungen (str)
 - Kontoauszug digital erhalten (bool)
- Documents
 - Ca. 20 Seiten zu Kontotypen, steuerlichen Hinweisen, Einkommensnachweis, AML-Richtlinien.
- Mögliche Tools
 - RAG für Abruf Kontotyp-Infos, Steuerregelungen, PEP-Infos
 - Alter-Rechner (aus Geburtsdatum)
 - Einkommen-zu-Kontotyp-Vergleichs-Tool
- Mögliche User:innen-Nachfrage-Prompts
 - „Welche Vorteile hat ein Girokonto?“ -> RAG
 - „Bin ich mit Einkommen X berechtigt, ein Premium-Konto zu eröffnen?“ -> Einkommen-Vergleich
 - „Was bedeutet politisch exponierte Person (PEP)?“ -> RAG
 - „Wie alt bin ich, wenn mein Geburtsdatum der 12.03.1985 ist?“ -> Alter-Rechner
 - „Formuliere bitte die Quelle der Einzahlungen höflicher als ‚kommt aus Arbeit‘.“ -> Kein Tool-Aufruf

UC4. Versicherungsberatungs-Assistent

- Felder
 - Versicherungstyp (str)
 - Alter Versicherungsnehmer (int)
 - Schadenfreiheitsklasse (Optional[int])
 - Vorversicherung vorhanden (bool)
 - Versicherungswert (€) (Optional[int])
 - Selbstbeteiligung (€) (int)
 - Risikogebiet (bool)
 - Laufzeit in Jahren (int)
 - Gesundheitsfragen beantwortet (bool)
 - Wunsch Zusatzoptionen (Optional[str])
- Documents
 - 20 Seiten zu verschiedenen Versicherungen, Schadenfreiheitsklassen, Zusatzoptionen, Gesundheitsfragen.
- Mögliche Tools
 - RAG für Versicherungsinfos, Schadenfreiheitsklassen
 - Risiko-Rechner (auf Basis von Gebiet)
 - Laufzeit-Kosten-Rechner
- Mögliche User:innen-Nachfrage-Prompts
 - „Welche Schadenfreiheitsklasse ist nach 5 unfallfreien Jahren üblich?“ -> RAG
 - „Ist mein Gebiet XY als Risikogebiet eingestuft?“ -> Risiko-Rechner
 - „Welche Zusatzoptionen gibt es für eine KFZ-Versicherung?“ -> RAG
 - „Wie verändert sich der Preis bei 10 Jahren Laufzeit gegenüber 5 Jahren?“ -> Laufzeit-Kosten-Rechner
 - „Hilf mir bei der Formulierung ‚möchte zusätzliche Leistungen‘ eleganter auszudrücken.“ -> Kein Tool-Aufruf

UC5. Reiseplanungs-Assistent

- Felder
 - Zielort (str)
 - Reisebeginn (str)
 - Dauer (int)
 - Budget (€) (int)
 - Anzahl Reisende (int)
 - Unterkunft gewünscht (bool)
 - Transportmittel (str)

- Flexible Reisedaten (bool)
- Aktivitäten vor Ort (Optional[str])
- Reiseversicherung (bool)
- Documents
 - 20 Seiten Reiseziele, Unterkünfte, Transportmöglichkeiten, Aktivitäten
- Mögliche Tools
 - RAG für Zielort-Infos
 - Budget-Planungsrechner
 - Wetter-API
 - Transportmittel-Vergleich
- Mögliche User:innen-Nachfrage-Prompts
 - „Welche Sehenswürdigkeiten gibt es in Paris?“ -> RAG
 - „Wie wird das Wetter im August in Barcelona?“ -> Wetter-API
 - „Was kostet eine durchschnittliche Unterkunft in Rom?“ -> RAG
 - „Kann ich mit meinem Budget von 1000€ für 5 Tage reisen?“ -> Budget-Planungsrechner
 - „Sag mir besser, was ich mit ‚Spaß-Aktivitäten‘ ausdrücken könnte.“ -> Kein Tool-Aufruf markdown Kopieren Bearbeiten

UC6. Bauantrags-Assistent

- Felder
 - Bauprojekt-Art (str)
 - Grundstückfläche (int)
 - Gebäudehöhe geplant (float)
 - Grundstück Eigentümer (bool)
 - Schutzgebiet (bool)
 - Architekt beauftragt (bool)
 - Baubeginn (str)
 - Gewerbliche Nutzung (bool)
 - Barrierefreiheit (bool)
 - Energieeffizienzklasse (str)
- Documents
 - 20 Seiten Bauvorschriften, Schutzgebiete, Energieeffizienz-Anforderungen.
- Mögliche Tools
 - RAG für Bauvorschriften und Schutzgebiet-Infos
 - Grundstücksflächen-Rechner

- Energieeffizienz-Rechner
- Mögliche User:innen-Nachfrage-Prompts
 - „Welche Vorschriften gelten für Neubauten im Schutzgebiet XY?“ -> RAG
 - „Ist mein Grundstück in einem Schutzgebiet?“ -> RAG
 - „Welche Energieeffizienzklassen gibt es?“ -> RAG
 - „Berechne mir die Grundstücksfläche aus Länge 20m und Breite 30m.“ -> Grundstücksflächen-Rechner
 - „Bitte hilf mir ‚wird gewerblich genutzt‘ besser auszudrücken.“ -> Kein Tool-Aufruf

UC7. Studienbewerbungs-Assistent

- Felder
 - Gewünschter Studiengang (str)
 - Semesterbeginn (str)
 - Abschlussnote letztes Zeugnis (float)
 - Motivationsschreiben vorhanden (bool)
 - Studiengebühren-Finanzierung gesichert (bool)
 - Sprachkenntnisse vorhanden (bool)
 - Anzahl Empfehlungsschreiben (int)
 - Vorpraktika absolviert (bool)
 - Studium im Ausland geplant (bool)
 - Bewerbung auf Stipendium (bool)
- Documents
 - 20 Seiten zu Bewerbungsprozessen, Studiengängen, Voraussetzungen, Auswahlverfahren.
- Mögliche Tools
 - RAG für Infos zu Studiengängen, Fristen, Finanzierung
 - Noten-Umrechner (z. B. zwischen Systemen)
 - Stipendien-Finder
- Mögliche User:innen-Nachfrage-Prompts
 - „Was gehört alles ins Motivationsschreiben für Soziale Arbeit?“ -> RAG
 - „Wie rechne ich meine Note von 2,3 ins US-System um?“ -> Noten-Umrechner
 - „Welche Stipendien kann ich mit meinem Profil beantragen?“ -> RAG + Stipendien-Finder
 - „Wie viele Empfehlungsschreiben sind üblich bei Informatik?“ -> RAG
 - „Wie kann ich ‚ich habe viel Erfahrung mit Kindern‘ besser formulieren?“ -> Kein Tool-Aufruf

UC8. Fördermittel-Antragsassistent

- Felder
 - Förderprogramm (str)
 - Beantragte Summe (€) (int)
 - Unternehmenstyp (str)
 - Gründungsdatum (str)
 - Anzahl Mitarbeiter (int)
 - Projektstart geplant (str)
 - Innovationsgrad hoch (bool)

- Kooperationspartner involviert (bool)
- Finanzierungsplan liegt vor (bool)
- Umweltfreundliche Kriterien erfüllt (bool)
- Documents
 - 20 Seiten über Förderkriterien, Beispiele für erfolgreiche Anträge, Richtlinien pro Programm.
- Mögliche Tools
 - RAG für Programm-Infos und Beispielanträge
 - Gründungsalter-Rechner (Jahre)
 - Förderchancen-Checker (einfaches Bewertungstool)
- Mögliche User:innen-Nachfrage-Prompts
 - „Was fördert das Programm ‚Green Start‘ genau?“ -> RAG
 - „Ich habe am 01.04.2019 gegründet – wie alt ist das Unternehmen jetzt?“ -> Gründungsalter-Rechner
 - „Wie bewerte ich meinen Innovationsgrad realistisch?“ -> RAG + Förderchancen-Checker
 - „Was muss alles in einen Finanzierungsplan rein?“ -> RAG
 - „Hilf mir, ‚wir sind nachhaltig‘ besser zu formulieren.“ -> Kein Tool-Aufruf

UC9. Datenschutz-Compliance-Assistent

- Felder
 - Unternehmenstyp (str)
 - Datenschutzbeauftragter vorhanden (bool)
 - Anzahl Mitarbeiter mit Zugriff auf sensible Daten (int)
 - Datenverarbeitungsverzeichnis aktuell (bool)
 - DSGVO-Schulungen regelmäßig durchgeführt (bool)
 - Datenschutzverletzung in letzten 12 Monaten (bool)
 - Verarbeitete Datentypen (Optional[str])
 - Datenübermittlung in Drittländer (bool)
 - Löschkonzept vorhanden (bool)
 - Zustimmungserklärung standardisiert (bool)
- Documents
 - 20 Seiten interne Datenschutzleitlinien, Beispiele von DSGVO-Umsetzungen, Audit-Checklisten.
- Mögliche Tools
 - RAG für DSGVO-Interpretationen und Maßnahmenkataloge
 - Datenschutz-Audit-Checker (z. B. für Basis-Compliance)

- Datenkategorisierungs-Hilfe
- Mögliche User:innen-Nachfrage-Prompts
 - „Was zählt als sensible Daten?“ -> RAG
 - „Wie sieht ein aktuelles Verzeichnis der Verarbeitungstätigkeiten aus?“ -> RAG
 - „Was ist ein Beispiel für eine standardisierte Zustimmungserklärung?“ -> RAG
 - „Wie viele Mitarbeiter sind zu viel für keinen Datenschutzbeauftragten?“ -> RAG
 - „Formuliere ‚wir haben einen Löschprozess‘ präziser.“ -> Kein Tool-Aufruf

UC10. Steuererklärungs-Assistent

- Felder
 - Steuerjahr (int)
 - Einkunftsart (str)
 - Einkommen im betreffenden Jahr (€) (int)
 - Anzahl abzugsfähiger Positionen (Optional[int])
 - Pendlerpauschale beantragt (bool)
 - Familienstand (str)
 - Kinderanzahl (int)
 - Altersvorsorge vorhanden (bool)
 - Einkünfte aus dem Ausland (bool)
 - Steuerberatung genutzt (bool)
- Documents
 - 20 Seiten Steuerleitfäden, Abzugspositionen, Beispiele von Formularen, Fristen.
- Mögliche Tools
 - RAG für steuerrechtliche Erläuterungen und Abzugsmöglichkeiten
 - Steuerklassen-Erklärer
 - Pendlerpauschale-Rechner
- Mögliche User:innen-Nachfrage-Prompts
 - „Was zählt als abzugsfähige Position?“ -> RAG
 - „Wie berechne ich die Pendlerpauschale für 20 km Arbeitsweg?“ -> Pendlerpauschale-Rechner
 - „Was bedeutet Steuerklasse IV mit Faktor?“ -> Steuerklassen-Erklärer
 - „Wie werden ausländische Einkünfte angegeben?“ -> RAG
 - „Bitte hilf mir ‚ich nutze eine Steuerberatung‘ besser zu schreiben.“ -> Kein Tool-Aufruf

UC11. Patent-Anmeldeassistent

- Felder
 - Titel der Erfindung (str)
 - Anmeldedatum geplant (str)
 - Erfinder (str)
 - Anzahl Co-Erfinder (int)
 - Erfindung bereits veröffentlicht (bool)
 - Gebiet der Technik (str)
 - Prioritätsanmeldung in anderem Land (bool)
 - Beschreibung verfügbar (bool)

- Anmelder Firma (bool)
- Recherche zu Neuheit durchgeführt (bool)
- Documents
 - 20 Seiten zu Patentanmeldung, Veröffentlichungsregeln, Prioritätsrechten, Formulierungsbeispielen.
- Mögliche Tools
 - RAG für Richtlinien und Vorlagen
 - Prioritätsfrist-Rechner
 - Technologiefeld-Zuordnungshilfe
- Mögliche User:innen-Nachfrage-Prompts
 - „Was sind die Voraussetzungen für eine Patentanmeldung?“ -> RAG
 - „Ich habe am 01.01.2025 veröffentlicht – bis wann kann ich anmelden?“ -> Prioritätsfrist-Rechner
 - „In welchem Technologiefeld liegt eine smarte Wasserflasche?“ -> RAG + Zuordnungshilfe
 - „Was bedeutet Prioritätsanmeldung?“ -> RAG
 - „Formuliere ‚noch keine Recherche gemacht‘ diplomatischer.“ -> Kein Tool-Aufruf

UC12. Lebensmittel-Beratungsassistent

- Felder
 - Alter (int)
 - Geschlecht (str, z.B. männlich, weiblich, divers)
 - Körpergewicht in kg (float)
 - Körpergröße in cm (float)
 - Allergien oder Unverträglichkeiten (Optional[str])
 - Essgewohnheiten (str, z.B. vegetarisch, vegan, omnivor, paleo)
 - Ziel (str, z.B. abnehmen, Muskelaufbau, Gewicht halten, fitter werden)
 - Körperliche Aktivität (str, z.B. wenig, moderat, sportlich)
 - Zeitaufwand für Kochen pro Tag in Minuten (int)
 - Gesundheitszustand / Vorerkrankungen (Optional[str])
- Documents
 - 20 Seiten mit Infos zu Diätformen, Ernährungsempfehlungen, Kontraindikationen (z. B. Low Carb bei Nierenproblemen), Beispiel-Tagespläne für verschiedene Ernährungstypen.
- Mögliche Tools
 - RAG für Abruf passender Diäten und Beispielpläne

- BMI-Rechner
- Kalorienbedarfsrechner
- Allergie-Check-Tool
- Diät-Vergleichstool nach Ziel und Einschränkungen
- Mögliche User:innen-Nachfrage-Prompts
 - „Was passt besser zu mir: Low Carb oder Mittelmeerdiet?“ -> RAG + Diät-Vergleich
 - „Ich bin 1,75 m groß und wiege 95 kg – wie hoch ist mein BMI?“ -> BMI-Rechner
 - „Wie viele Kalorien brauche ich bei wenig Bewegung?“ -> Kalorienbedarfsrechner
 - „Ich bin laktoseintolerant – welche Diäten schließen Milchprodukte aus?“ -> RAG + Allergie-Check
 - „Kannst du mein Ziel ‚fitter werden‘ klarer formulieren?“ -> Kein Tool-Aufruf

1B) Im Forums-Thread Auswahl mitteilen - First Come First Serve

<https://moodle.hochschule-burgenland.at/mod/forum/discuss.php?d=77135>

2. RAG-Komponente aufbauen

2A) Dokumente (ca. 20 Seiten) zusammenstellen

- Was z.B.?
 - Webseiten to PDFs
 - selbst kreierte Texte
 - von KI-Chat-Anwendung synthetisch erzeugte Texte
 - im Web gefundene PDFs
- Wie?
 - wenn mehrere Doks, alle in ein 20-Seiten-PDF mergen
- Auf was achten?
 - hier an Felder denken, was nützlich sein könnte für Befüllung oder Beratung oder Beispielgebung

2B) 20-Seiten-PDF prozessieren (Parsing, Chunking)

2C) Chunks einbetten und in kreierte Vektor-DB (Milvus Lite oder Andere) indexieren (Embedding, Indexing)

3. Agents und Tools ausprobieren

Beispielcode

```
In [ ]: # Pydantic Felder
class SupportDevice(BaseModel):
    device_type: str = Field(description="Type of device (e.g., Printer, Laptop)")
    model_number: str = Field(description="Model number of the device")
    os_version: str = Field(description="Operating system version")
    error_code: Optional[str] = Field(None, description="Error code if applicable")
    recurring_error: bool = Field(description="Whether the error occurs regularly")
    battery_life_hours: Optional[int] = Field(None, description="Battery life in hours")
    last_working_date: str = Field(description="Date when the device last worked")
    device_restarted: bool = Field(description="Whether the device has been restarted")
    issue_description: str = Field(description="Detailed description of the issue")
    warranty_valid: bool = Field(description="Whether the warranty is still valid")
```

```
In [ ]: # FileSearchTool Beispiel mit Anbindung an Vector DB (hier https://openai.github.io/milvus)

from agents import Agent, Runner, FileSearchTool

# Instantiate the file search tool with your Milvus Lite vector store id.
file_search_tool = FileSearchTool(
    vector_store_ids=["milvus_lite_store_id"], # Replace with your actual vector store id
    max_num_results=5, # Maximum number of results
    include_search_results=True # Include search results in the response
    # ranking_options and filters can be set here if needed
)

# Create an agent that uses the FileSearchTool.
agent = Agent(
    name="File Search Agent",
    instructions="You are a helpful assistant that can search for documents. When a query is received, use the FileSearchTool to find the relevant documents.",
    tools=[file_search_tool],
    model="gpt-4o" # Ensure you use a supported OpenAI model (e.g., GPT-4o)
)

# Run the agent with a sample query.
response = Runner.run(agent, "Find documents related to machine learning research")
print(response)
```

Gutes Gelingen! :)