



# KNIME AI Extension Guide

KNIME AG, Zürich, Schweiz

Version 5.7 (letzte Aktualisierung auf )



## Inhaltsverzeichnis

<a href="#page2" style="color: #000000; text-decoration: underline;">Ich prompiere ein Modell [#page2](#)  
<a href="#page2" style="color: #000000; text-decoration: underline;">Installieren Sie die KI [#page2](#)  
<a href="#page3" style="color: #000000; text-decoration: underline;">Authenticate . . . . . <a href="#">#page3  
<a href="#page5" style="color: #000000; text-decoration: underline;">Wählen Sie . . . . . <a href="#">#page5  
<a href="#page5" style="color: #000000; text-decoration: underline;">. . . . . <a href="#">#page5  
<a href="#page18" style="color: #000000; text-decoration: underline;">Anbieter Referenz [#page18](#)  
<a href="#page20" style="color: #000000; text-decoration: underline;">Retrieval-Augmented Generation [#page20](#)  
<a href="#page20" style="color: #000000; text-decoration: underline;">Verbinden Sie mit [#page20](#)  
<a href="#page21" style="color: #000000; text-decoration: underline;">Vector Store erstellen [#page21](#)  
<a href="#page22" style="color: #000000; text-decoration: underline;">. . . . . <a href="#">#page22  
<a href="#page22" style="color: #000000; text-decoration: underline;">Antwort generieren [#page22](#)  
<a href="#page23" style="color: #000000; text-decoration: underline;">Beispiel: Product [#page23](#)  
<a href="#page29" style="color: #000000; text-decoration: underline;">Agentische KI in KI [#page29](#)  
<a href="#page30" style="color: #000000; text-decoration: underline;">Verbinden Sie mit [#page30](#)  
<a href="#page31" style="color: #000000; text-decoration: underline;">Zugriffstools . . . <a href="#">#page31  
<a href="#page31" style="color: #000000; text-decoration: underline;">. . . . . <a href="#">#page31  
<a href="#page34" style="color: #000000; text-decoration: underline;">Eingabedaten . . . <a href="#">#page34  
<a href="#page34" style="color: #000000; text-decoration: underline;">Wie erstellt man [#page34](#)  
<a href="#page40" style="color: #000000; text-decoration: underline;">Checkliste: Was Sie brauchen [#page40](#)  
<a href="#page41" style="color: #000000; text-decoration: underline;">Beispiel: Bauen Sie einen [#page41](#)  
<a href="#page60" style="color: #000000; text-decoration: underline;">AI Governance. . . . <a href="#">#page60  
<a href="#page60" style="color: #000000; text-decoration: underline;">GPT4All (Lokale Modelle) [#page60](#)

# Ein Modell abgeben

Dieser Abschnitt des Leitfadens erklärt, wie man eine Aufforderung an ein Large Language Model (LLM) in KNIME Analytics Platform.

Das Verfahren folgt drei Schritten:

- [Authenticate](#page3)
- [Wählen](#page5)
- [Prompt](#page5)

Um jeden Schritt zu klären, enthält dieser Abschnitt eine [Beispiel-Workflow](#). Der Workflow verbindet sich mit OpenAI GPT-4.1 Modell und fasst Produktbewertungen in einem .csv Datei.

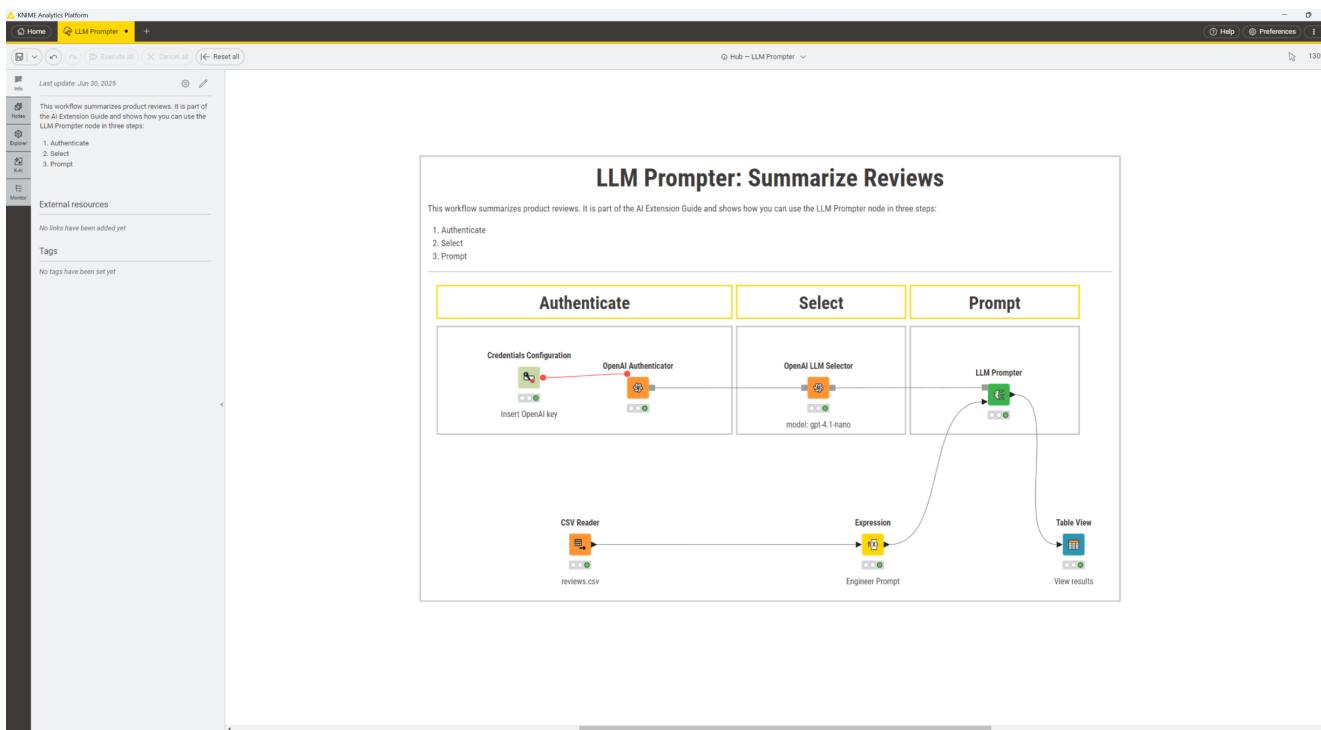


Abbildung 1. Beispiel-Workflow mit OpenAI GPT-4.1 in KNIME zur Zusammenfassung der Produktbewertungen über Authentifizierung, Modellauswahl und Aufforderung.

Installieren Sie die KNIME AI Extension

Um AI in KNIME Workflows zu verwenden, installieren Sie die [KNIME AI Erweiterung](#). Sie können dies auf zwei Arten tun:

- Von KNIME Hub: [KNIME AI Erweiterung](#) von [KNIME Hubraum](#) in Ihr KNIME Arbeitsraum.
- Aus der KNIME Analytics Plattform:

- a. Gehen Sie Menü schließen aus der Symbolleiste.
- B. Wählen Erweiterungen installieren.
- c. Suchen Sie nach "AI Extension" und folgen Sie den Anweisungen, um die Installation abzuschließen.

## Authenticate

Um Aufforderungen an Modelle zu senden, müssen Sie mit Ihrem gewählten Modellanbieter authentifizieren.

Die meisten Anbieter benötigen einen API-Schlüssel oder Token aus Ihrem Benutzerkonto.

Authentication umfasst typischerweise zwei Knoten:

- [Konfigurieren von Anmeldeinformationen](#)
- [Authenticator \(z. \[OpenAI Authentication\]\(#\)\)](#)

### [Anmeldeinformationen Konfigurationsknoten](#)

Dieser Knoten speichert Anmeldeinformationen als Strömlungsgröße für nachgeschaltete Knoten.

In KNIME, a **Durchflussgröße** ist ein benannter Wert, der Daten wie Anmeldeinformationen oder Konfiguration übergibt Einstellungen, zwischen Knoten. Durch die Verwendung von Durchflussgrößen wird der Workflow flexibler und vermeidet heikle Informationen.

Um den Knoten zu konfigurieren:

- Geben Sie die API-Taste oder das Token in die Passwort vergessen Feld.
- Lassen Sie die Benutzername Feld leer
- Wenn Sie nicht überprüfen Passwort in der Konfiguration speichern (schwach verschlüsselt) , der Schlüssel wird nicht gespeichert zwischen Sitzungen und muss beim Wiedereröffnen des Workflows erneut eingeschleust werden.

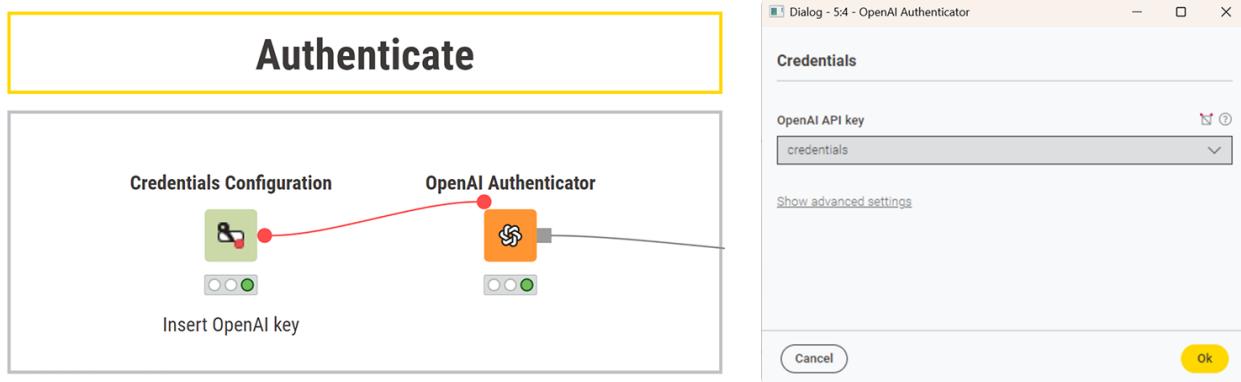


Abbildung 2. Die [Konfigurieren von Anmeldeinformationen](#) node speichert die OpenAI API-Taste im Passwort-Feld und erstellt eine Anmeldedatenflussvariable für den Einsatz in nachgeschalteten Authentifizierungsknoten.

#### Authenticator Node

Den Anmeldeinformationen zuordnen Flussgröße dem

API SchlüsselFeld im Authentisierungsknoten

Konfiguration. anbieterspezifische Parameter (z.  
erforderlich.

Gebiet oder Endpunkt ) kann auch

Wenn die Anmeldeinformationen ungültig oder unvollständig sind, wird der Authentisierungsknoten während der Ausführung ausfallen.

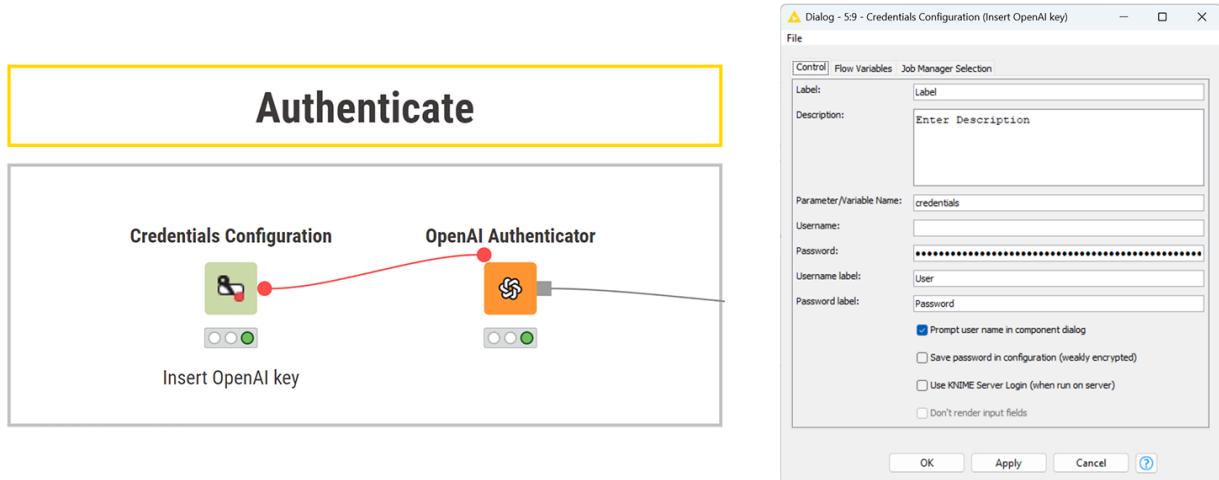


Abbildung 3. In diesem Beispiel [OpenAI Authentication](#) node ist konfiguriert, um Anmeldeinformationen abzurufen aus der von der [Konfigurieren von Anmeldeinformationen](#) Knoten.

## Wählen

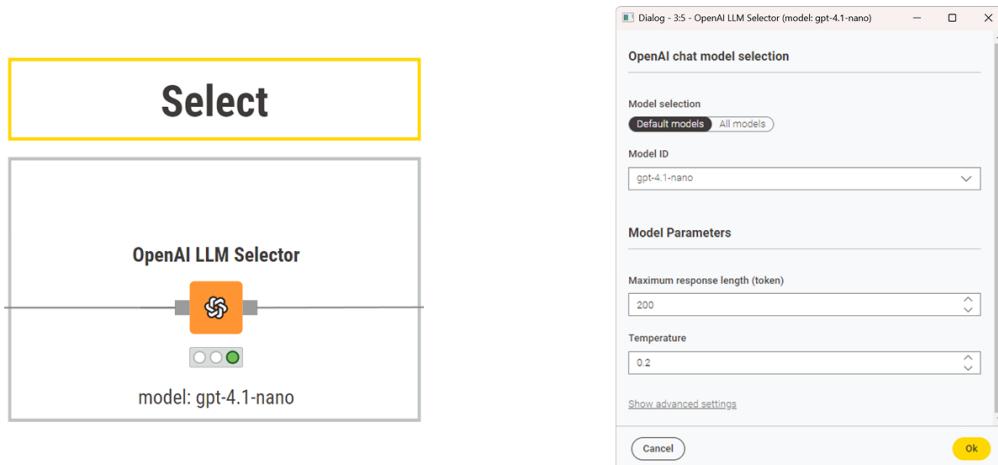
Verwenden Sie nach der Authentisierung den entsprechenden Connectorknoten, um das gewünschte Modell auszuwählen Bitte.

Mit den Selector-Knoten können Sie:

- Modelle aus kommerziellen APIs auswählen ( [OpenAI](#) , [Gefällt mir](#), [Anthropin](#) , usw.)
- Verbinden Sie mit Modellen, die auf dem KNIME Hub gehostet werden, oder laufen Sie lokal über GPT4All.

### Modellauswahlparameter

- Max neue Token / Max Response Länge: maximale Antwortgröße.
- Temperatur : steuert Ausgabe-Zufälligkeit (0 = deterministisch, höher = kreativer).
- Erweiterte Einstellungen: zusätzliche Parameter wie Top-p Probenahme , Siehe , oder Parallel Anträge .



**Abbildung 4.** Die [OpenAI LLM Selector](#) node ist konfiguriert, um das Modell GPT-4.1 von OpenAI zu verwenden Produktbewertungen zusammenfassen.

Für einen vollständigen Überblick über unterstützte Anbieter, verfügbare Authentisierungs- und Selektorknoten, benötigte Anmeldeinformationen, und zum Beispiel Workflows, siehe <#page18>.

## Prompt

Freigabe bedeutet, Textanweisungen an ein Sprachmodell zu senden, um bestimmte Aufgaben auszuführen. Abhängig vom verwendeten Knoten kann das Modell Text erzeugen, Fragen beantworten, extrahieren eine Information oder Rückgabe von Vektordarstellungen der Eingabe.

In KNIME ist eine Aufforderung einfach eine Textzeile. Sie können diese Strings mit solchen Knoten erstellen wie [Ausdruck](#), [String Manipulation](#), [oder Tabelle Schöpfer](#).

Die [KNIME AI](#) Erweiterung verfügt über drei Eingabeknoten:

[`<a href="#page6" style="color: #ff6600; text-decoration: underline;">LLM Prompter</a>`](#)

Ein-Dreh-Text zur Eingabe

- [\[LLM Chat Prompter\]](#)

Chat-Stil Multi-turn-Anforderungen

[`<a href="#page14" style="color: #ff6600; text-decoration: underline;">Text einbetten</a>`](#)

Generate Einbettungen (Vektordarstellungen)

## LLM Prompter

Die [LLM Prompter](#) node sendet einfache Textansagen an ein Sprachmodell und gibt das

Die Antwort des Modells als Text. Es wird für Ein-Schuss-Einleitung verwendet, die nicht benötigt Geschichte.

Allgemeine Anwendungsfälle:

- Klassifizierung
- Verschmelzung
- Umschreiben
- Extraktion

### [Beispiel: Summarize Produktbewertungen](#)

Sie erhalten Produktbewertungen, die zu lengthy sind, so dass Sie sich entscheiden, sie zusammenzufassen. Der Eingang Daten sehen so aus:

ID	Anmerkung
1	Das Produkt kam pünktlich an... benutzerfreundlich.
2.	Das Produkt kam pünktlich an... jeden Tag.
3	Ich habe das als Geschenk gekauft... andere Familienangehörige.

## **ANHANG Lesen Sie die Daten**

Verwenden Sie die [CSV Reader](#) Knoten, um diese Tabelle in Ihren KNIME Workflow zu laden. Jede Überprüfung wird gespeichert als String in der Kommentarspalte.

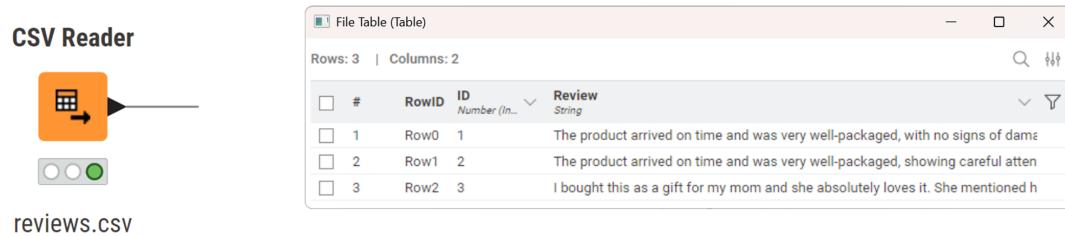


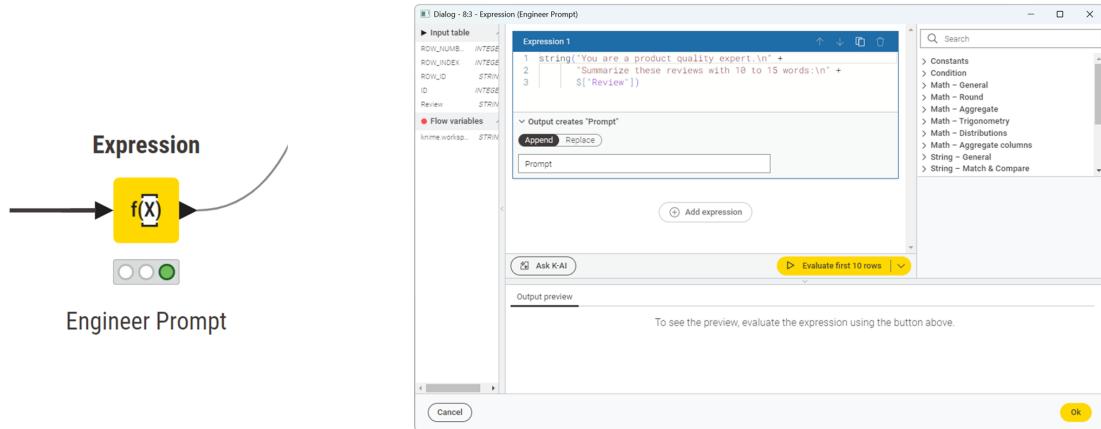
Abbildung 5. Die [CSV Reader](#) node lädt eine Tabelle der Kundenrezensionen, die jeweils als Zeichenkette gespeichert sind, später zusammengefasst werden.

## 2. Die Eingabeaufforderungen erstellen

Verwenden Sie die **Ausdruck** um diese Aufforderung zu erstellen:

string("Sie sind ein Produktqualitätsexperte.\n" +  
"Summarisieren Sie diese Bewertungen mit 10 bis 15 Zeichen:\n" +  
\$["Review"]

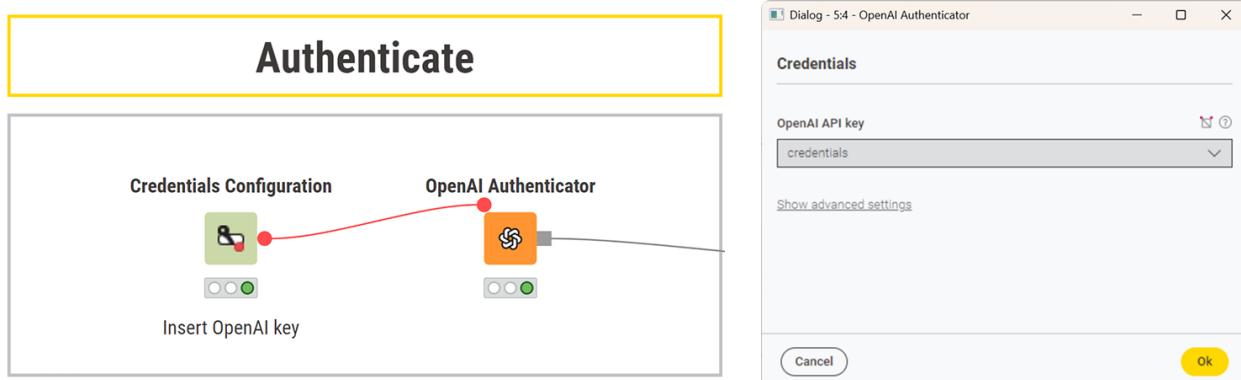
Dies schafft eine neue Spalte namens Bitte die den Befehl für jede Zeile enthält.



**Abbildung 6.** Die [Ausdruck](#) node erstellt eine neue Spalte namens Aufforderung, die die Anweisungen für die LLM zu folgen.

### 3. Authenticate to OpenAI

Verwenden Sie die [Konfigurieren von Anmeldeinformationen](#) und [OpenAI Authentication](#) Node, um Ihre API Schlüssel.



**Abbildung 7.** Die [Konfigurieren von Anmeldeinformationen](#) speichert sicher Ihren API-Schlüssel. Die [OpenAI Authentication](#) node liest diesen Schlüssel, um Anfragen an die OpenAI API zu autorisieren.

L 347 vom 20.12.2013, S. 1). Auswahl eines Modells

Verwenden Sie die [OpenAI LLM Selector](#) node, um das gewünschte Modell zu wählen.

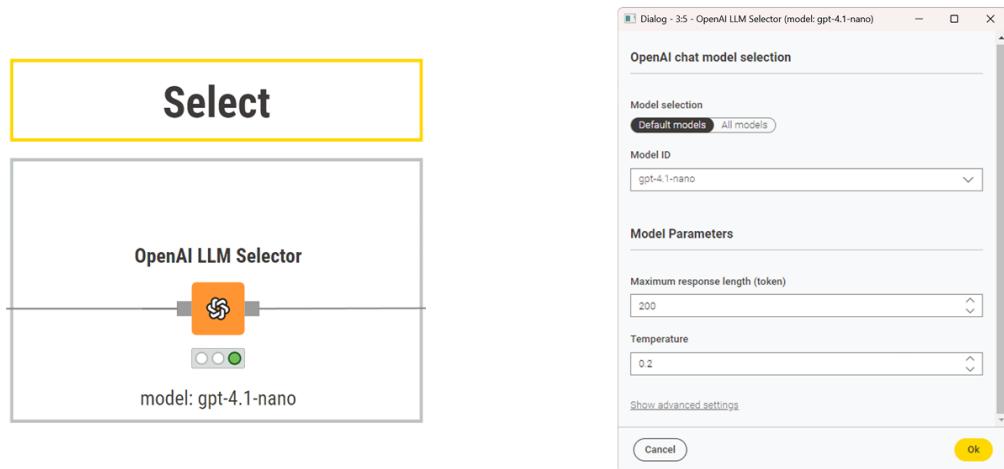


Abbildung 8. Die [OpenAI LLM Selector](#) node verbindet mit der API und wählt GPT4 als Modell für Ich lade Sie ein.

## 5. Prompt

Die [LLM Promter](#) knoten liest die Prompt Spalte, sendet es an das Modell, und speichert die Antwort des Modells in einer neuen Spalte namens Antwort .

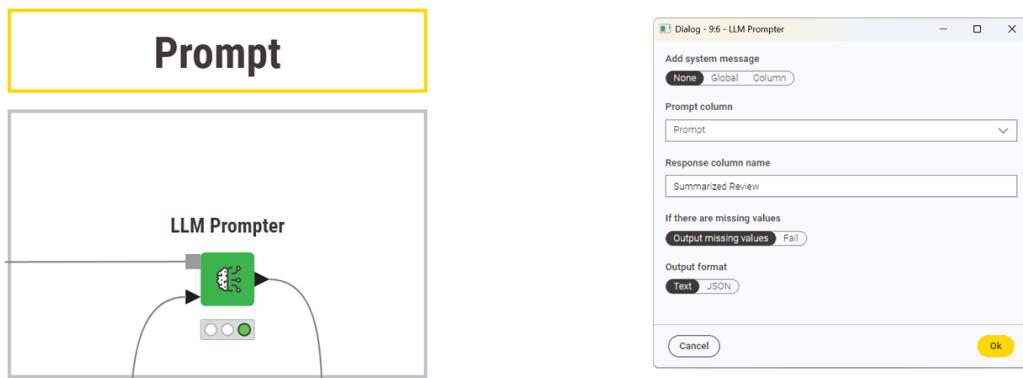


Abbildung 9. Die [LLM Promter](#) node verwendet die angegebene Eingabeaufforderungsspalte und erzeugt eine Antwort aus dem Modell, das es in einer neuen Spalte namens Antwort speichert.

Um strukturierte JSON Antworten zu erhalten, stellen Sie sicher, dass das ausgewählte Modell JSON-Modus unterstützt.

Auch, wählen JSON als Ausgabeformat in der Knotenkonfiguration und explizit anfordern

JSON Format in der Aufforderung selbst. In vielen Fällen hilft es, einige Beispiele der erwartete Struktur direkt in der Aufforderung, die Ausgabequalität zu gewährleisten.

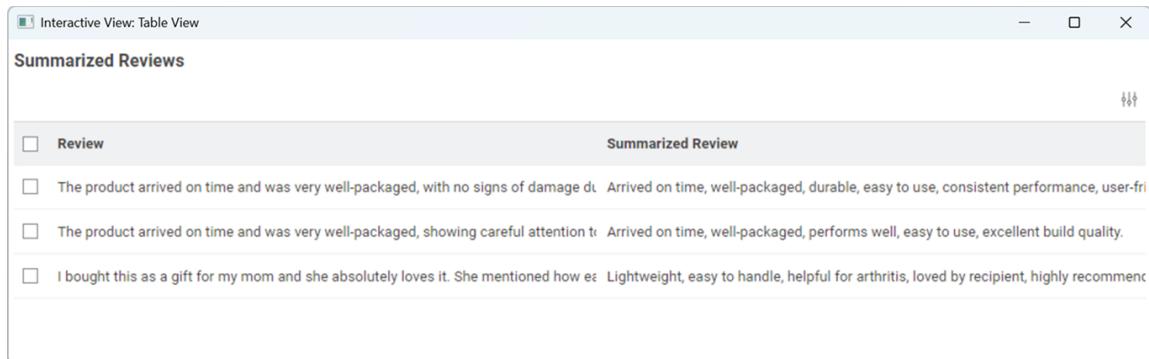


Abbildung 10. Am Ende des Arbeitsablaufs, [LLM Prompter](#) node gibt eine neue Spalte aus

Antwort. Jede Zeile enthält eine Zusammenfassung, die vom Modell anhand der entsprechenden Kundenbewertung.

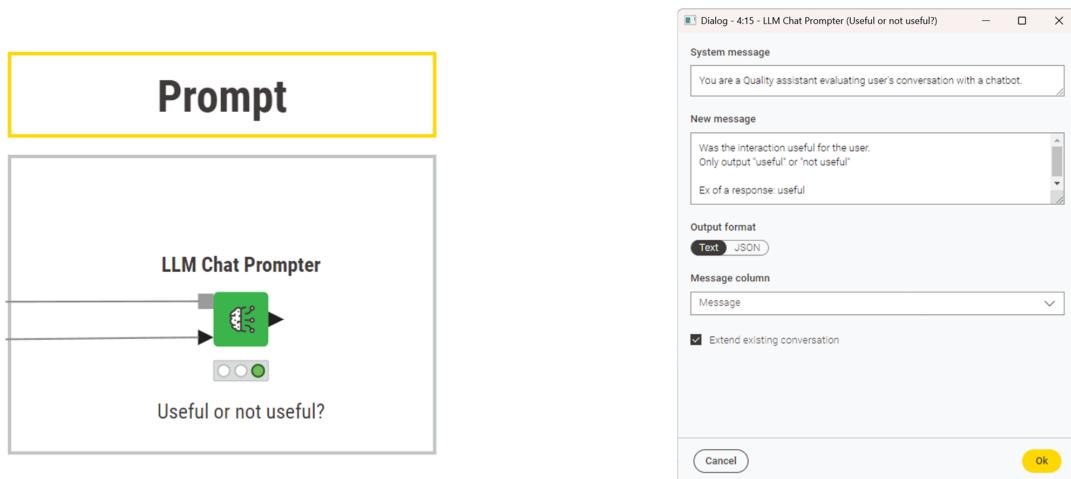
### [LLM Chatten Sie ein](#)

Die [LLM Chatten Sie ein](#) node ermöglicht Chat-style-Anforderungen an ein Sprachmodell. Im Gegensatz zu [LLM Promoter](#), es akzeptiert Gesprächsgeschichte als Eingabe, um Kontext für die Generierung Antworten.

Die [LLM Chatten Sie ein](#) node verwaltet nicht automatisch Multiturn Gespräche. Die vollständige Konversationsgeschichte muss als Eingabe für jede Ausführung, wenn der vorherige Kontext erforderlich ist.

### Konfiguration

- Systemnachricht (optional): : definiert Assistenzverhalten, z. "Sie sind ein hilfreicher Kunde Support Agent."
- Neue Benutzernachricht (optional): eine neue Benutzernachricht angibt.
- Ausgabeformat: Klartext oder JSON.
- Nachricht Column: Spalte mit KNIME Nachrichtentyp (Rolle, Nachricht).



**Abbildung 11.** Konfigurationsdialog der [LLM Chatten Sie ein](#) Knoten.

#### Eingänge

Der Knoten akzeptiert zwei zusätzliche Eingangssports:

- Konversationsgeschichte (optional):

Eine Tabelle mit vorherigen Gesprächsnachrichten im Nachrichtenformat von KNIME. Diese Nachrichten werden als Kontext für die aktuelle Aufforderung verwendet. Der Knoten nicht automatisch Multiturn-Sitzungen verwalten. Die volle Geschichte muss aufrechterhalten werden extern und bei jeder Knotenausführung wieder bereitgestellt, wenn ein vorheriger Kontext erforderlich ist.

- Werkzeugdefinitionen (optional, erweitert):

Eine Tabelle mit Werkzeugdefinitionen im JSON-Format. Diese Definitionen ermöglichen Werkzeuganruf Funktionalität. Werkzeugdefinitionen werden im Workflow-Bereich von Agents detailliert erläutert. diese Anleitung.

#### [Beispiel: Qualitätskontrolle für Kunden](#)

Sie möchten ein Gespräch zwischen einem Benutzer und einer KI auswerten. Die Eingabedaten sehen so aus:

Rolle	Nachricht
e)	
Benutzer	Mein Internet trennt sich immer wieder.
ai	Es tut mir leid für die Unannehmlichkeiten. Haben Sie versucht, Ihren Router neu zu starten?

Benutzer	Ja, ich habe es mehrmals neu gestartet.
ai	Verstanden. Sind alle Anzeigeleuchten auf Ihrem Router normal?
Benutzer	Ja, alles sieht gut aus.
ai	Danke für die Bestätigung. Es könnte ein Line-Problem sein. Ich empfehle die Überprüfung mit Ihrem Internet-Anbieter.
Benutzer	Ich kontaktierte sie schon, aber sie sagten, alles scheint auf ihrer Seite gut zu sein. Die Das Problem bleibt bestehen.

#### ANHANG Konversationsgeschichte erstellen

Die [Tabelle Schöpfer](#) node erstellt eine Gesprächstabelle mit zwei Spalten: Rolle und Text . Das simuliert frühere Wendungen zwischen dem Benutzer und dem AI Assistenten.

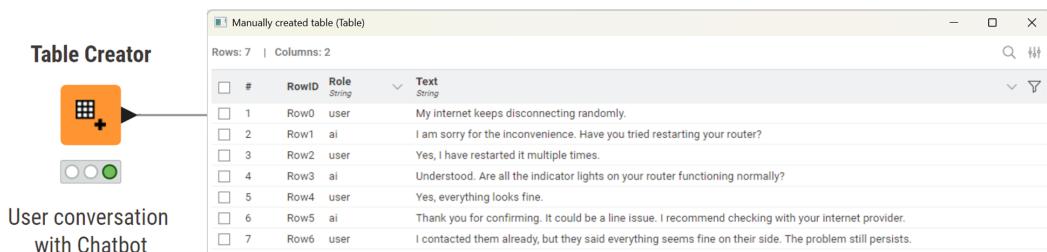


Abbildung 12. [Tabelle Schöpfer](#) node erstellt eine Tabelle mit zwei Spalten: role and message.

#### 2. Konvertieren zum Nachrichtentyp

Die [Nachricht Schöpfer](#) node transformiert die Gesprächstabelle in die Nachrichtendaten von KNIME Typ.

Die Eingabetabelle enthält bereits eine Spalte namens Rolle. Wählen Sie im Konfigurationsdialog diese aus Spalte unter Role Column.

Der Nachrichtentext befindet sich bereits in einer Spalte, die den Konversationsinhalt enthält. Wählen Sie diese Spalte unter Text.

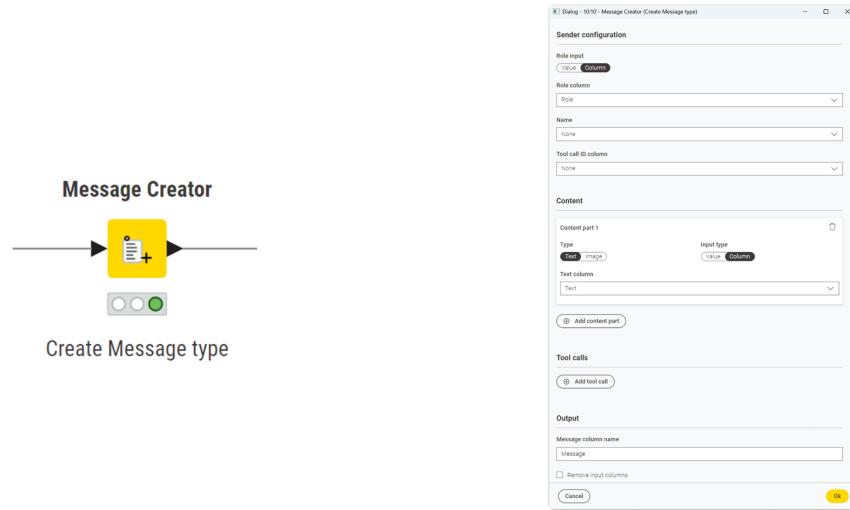


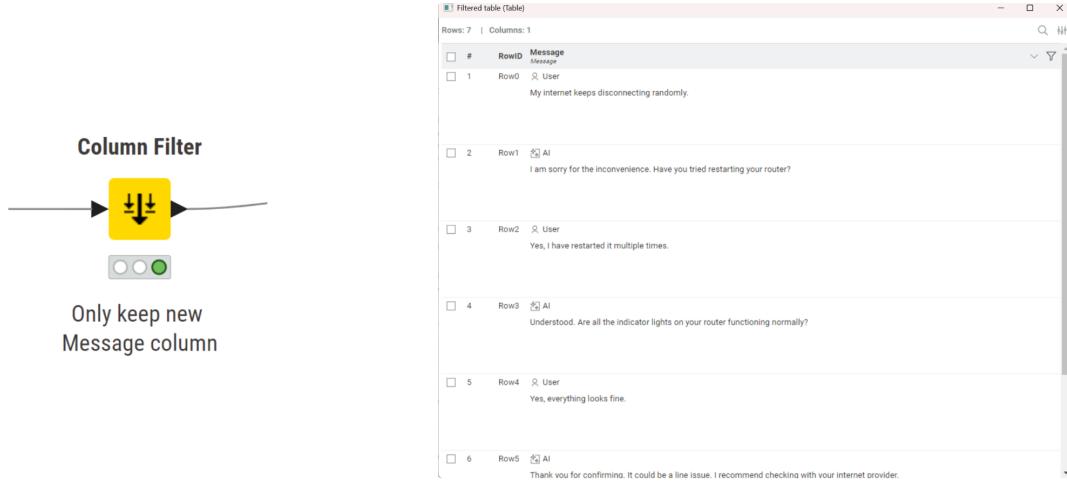
Abbildung 13. Konfigurationsdialog des Nachrichten-Ersteller-Knotens.

Nach der Konfiguration des Knotens erstellt dies eine neue Spalte namens Nachricht, die die KNIME Nachrichtentypen.

Diese Spalte kombiniert sowohl Rollen- als auch Nachrichteninhalte in einem Format, das nachgeschaltet benötigt wird [Knoten wie die LLM Chatten Sie ein](#).

### 3. Filter Nachrichtenspalte

Die [Spaltenfilter](#) node hält nur die Nachrichtenspalte für die Eingabe in die [LLM Chat Promoter](#).

Abbildung 14. Nachrichtenspalte in die [LLM Chatten Sie ein](#) Knoten.

### 4. Authenticate to OpenAI

Die [Einsteiger Widget](#) und [OpenAI Authentication](#) nodes verwalten Authentifizierung.

## 5. Modell auswählen

Die [OpenAI LLM Selector](#) Knoten selektiert gpt-4o-nano als LLM.

## 6. Prompt

Die [LLM Chatten Sie ein](#) node appends the new user an das bereitgestellte Gespräch

Geschichte und fordert eine Antwort vom Modell.

- Systemnachricht :

Sie sind ein Qualitätschecker für Kunden.

- Neue Benutzer-Nachricht :

Was die Interaktion für den Benutzer nützlich? Nur "brauchbar" oder "nicht nützlich" ausgegeben.

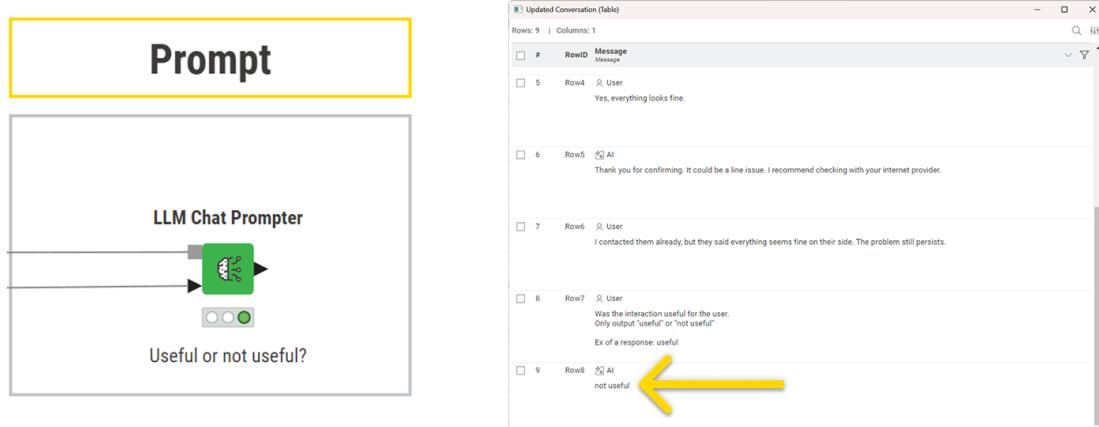


Abbildung 15.

[LLM Chatten Sie ein](#)

Knoten mit aktualisiertem Gespräch.

Text einbetten

Die [Text einbetten](#) node konvertiert Text in Einbettungen, die numerische Vektoren sind, die die Bedeutung des Textes erfassen.

Was ist ein Vektor?

Ein Vektor ist eine Liste von Zahlen, die einen Text im mathematischen Raum darstellen. Texte mit ähnlichen In diesem Raum werden die Bedeutungen eng zusammengelegt, auch wenn sie unterschiedliche Wörter verwenden. Für Beispiel: Hund und Welpen würde von nahe gelegenen Vektoren dargestellt werden. Einbettungen ermöglichen Modelle

die Bedeutung, die Gruppe ähnliche Texte und die Suche auf der Grundlage der Semantik anstatt einfach zu vergleichen Keyword-Anpassung.

### Keine Ausgabe

Der Knoten verarbeitet jede Zeile der Eingabetabelle und erstellt eine neue Vektorsäule, die die Einbettung. Jede Einbettung enthält in der Regel Hunderte oder Tausende von numerischen Werte. Im Gegensatz zu [LLM Promoter](#) Knoten (die natürlichen Sprachtext erzeugen), die [Text Einbettung](#) erzeugt rein numerische Einbettungen, voll kompatibel mit Abstandsberechnungen, Clustering- und Dimensionsreduktionsoperationen in KNIME.

### Allgemeine Anwendungsfälle:

- Semantische Suche: Texte mit ähnlicher Bedeutung finden
- Clustering: Texte in Kategorien gruppieren
- Ähnlichkeits-Scoring: Messen, wie nahe zwei Texte im Sinne stehen

### [Beispiel: Job Candidate Ähnlichkeit Plot](#)

Sie möchten den besten passenden Kandidat für eine Machine Learning-Spezialposition finden, die erfordert ein tiefes Wissen über NLP, Verstärkungslernen und Erfahrung mit großen Sprachmodellen.

Die Eingabedaten sehen so aus:

Kandidaten	Lebenslauf
Kandidaten 1	Senior Data Scientist mit 5 Jahren in NLP und Deep Learning. Erfahrungen in Python, TensorFlow und Cloud Computing.
Kandidaten 2.	Business Analyst mit Kompetenz in Marktforschung, Excel und Kunden Einsichten. Begrenzte Programmiererfahrung.
Kandidaten 3	ETL-Pipelines, Big Data Processing, Spark und verteilte Systeme.
Kandidaten ANH ANG	Maschinenbauer spezialisiert auf natürliches Sprachverständnis, Stärkung des Lernens und generative Modelle.
Kandidaten 5.	Software-Entwickler mit starkem Hintergrund in Java, Web-Entwicklung und Full-Stack Anwendungen.

Kandidaten	Lebenslauf
Ideal	Auf der Suche nach einem Spezialisten für maschinelles Lernen mit tiefem Wissen über NLP,
Profil	Verstärkungslernen und Erfahrung mit großen Sprachmodellen.

**ANHANG Lesen Sie die Daten**

Die [CSV Reader](#) node lädt die Tabelle der CVs und das ideale Profil in KNIME.

**2. Authenticate**

Die [Konfigurieren von Anmeldeinformationen](#) node speichert die OpenAI API-Taste. Die [OpenAI Authentication](#) node verwendet diese Anmeldeinformationen, um mit dem OpenAI-Service zu authentifizieren.

**3. Wählen**

Die [OpenAI Embedding Modellauswahl](#) node wählt das Einbettungsmodell [Text-Embedding-3](#) klein die verwendet werden, um Einbettungen zu erzeugen.

**4. Prompt (Generate Einbettungen)**

Die [Text einbetten](#) node konvertiert jeden CV-Text in einen Einbettungsvektor mit dem ausgewählten Einbettungsmodell.

**5. Abmessungen reduzieren**

Einbettungen sind hochdimensionale Vektoren, die nicht direkt aufgetragen werden können. Die [Sammlung](#) [Spalte](#) Knoten trennt den Vektor in einzelne numerische Spalten. Dann die [PCA](#) Knotenpunkt reduziert die vielen Abmessungen bis zu zwei, so dass die Einbettungen für 2D geeignet Visualisierung.

**6. Plot Ergebnisse**

Die [Scatter Plot](#) node zeigt, wie genau jeder Kandidat das ideale Profil passt auf semantische Ähnlichkeit.

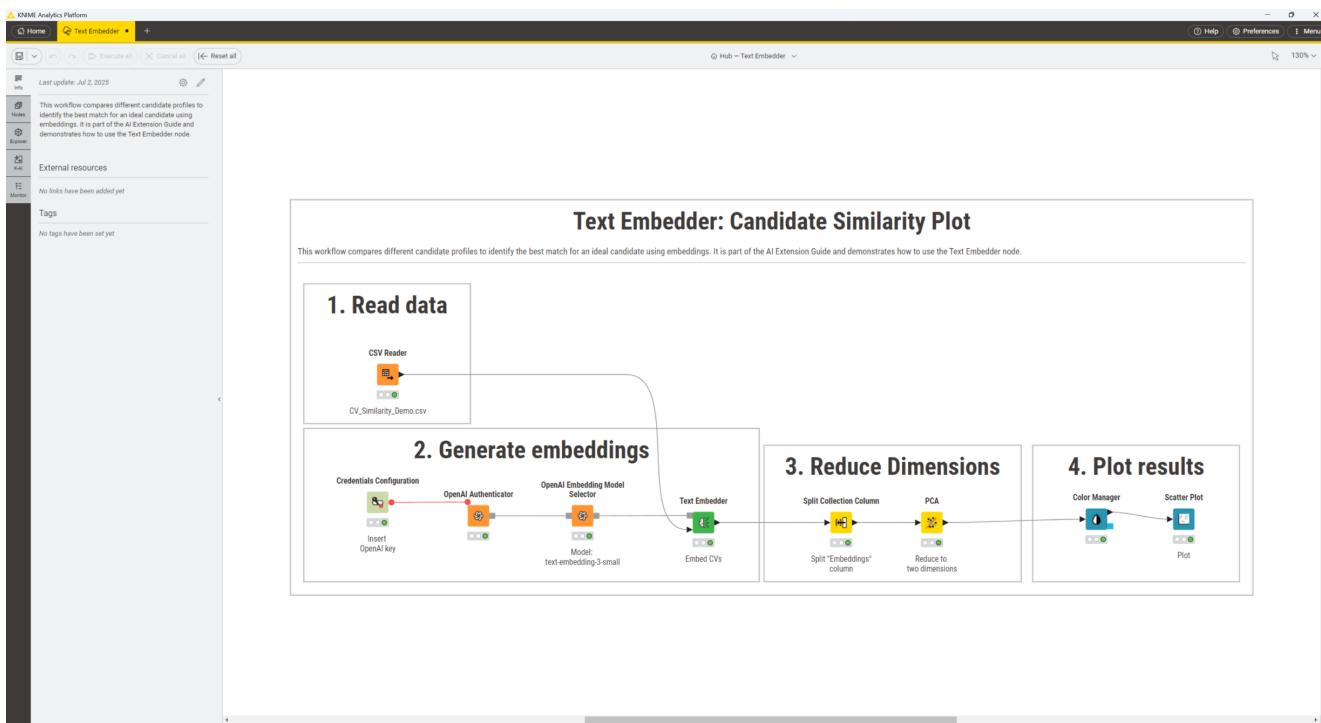


Abbildung 16. Der KNIME Workflow, der Kandidatenprofile mit Einbettungen und Plots vergleicht

die Ergebnisse

Das Grundstück zeigt, dass Kandidaten 4 ist dem idealen Profil am nächsten, bestätigt semantische Ähnlichkeit zwischen ihrer Erfahrung und dem Zielprofil.

Ideal Profil	Auf der Suche nach einem Spezialisten für maschinelles Lernen mit tiefem Wissen über NLP, Verstärkungslernen und Erfahrung mit großen Sprachmodellen.
Kandidaten ANH ANG	Maschinenbauer spezialisiert auf natürliches Sprachverständnis, Stärkung des Lernens und generative Modelle.

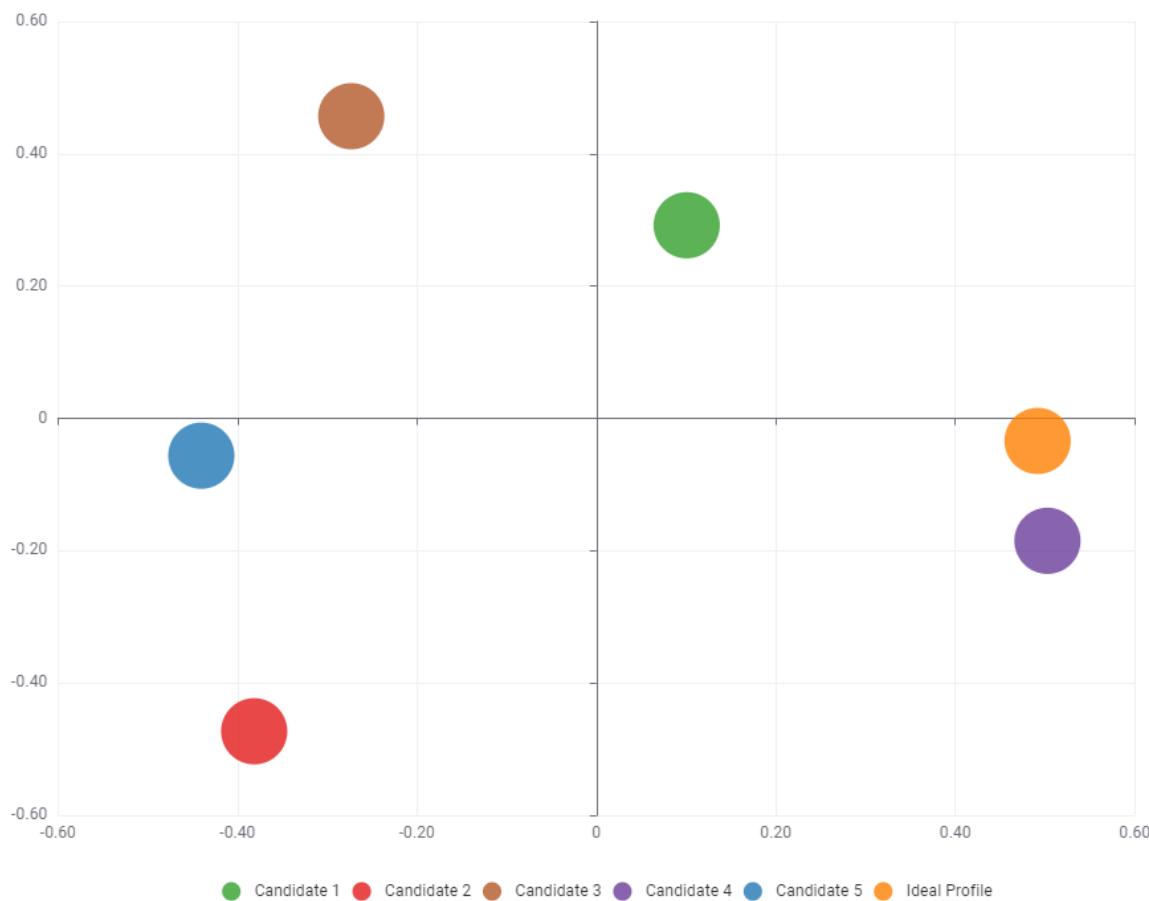
**Candidate Similarity Plot**

Abbildung 17. Das resultierende 2D-Plot der Kandidatähnlichkeit

**Referenztabelle des Anbieters**

Authentischer Knoten (von Anbieter)	Auswählen	Erforderlich Angaben	Link zu Anbieter	Examen Le Arbeitsblatt Eule
<a href="#">OpenAI</a> <a href="#">Authentischer</a>	<ul style="list-style-type: none"> <li>• <a href="#">OpenAI LLM Auswählen</a></li> <li>• <a href="#">OpenAI Einbettungen Modell Auswählen</a></li> </ul>	<ul style="list-style-type: none"> <li>• OpenAI API Schlüssel</li> <li>• OpenAI Basis URL (optional)</li> </ul>	<a href="#">OpenAI</a>	<a href="#">Arbeitsblatt Eule</a>

Authentischer Knoten (von Anbieter	Auswählen	Erforderlich Angaben	Link zu Anbieter	Examen Le Arbeitsblatt Eule
<a href="#">Google AI Studio Authentischer</a>	<ul style="list-style-type: none"> <li>• <a href="#">Gefällt mir</a> <a href="#">Auswählen</a></li> <li>• <a href="#">Gemini Einbettung</a> <a href="#">Modellauswahl</a></li> </ul>	<ul style="list-style-type: none"> <li>• Google AI Studio API Schlüssel</li> </ul>	<a href="#">Google</a>	<a href="#">Arbeitsblatt</a> <a href="#">Eule</a>
<a href="#">Anthropin Authentischer</a>	<ul style="list-style-type: none"> <li>• <a href="#">Anthropische LLM</a> <a href="#">Auswählen</a></li> </ul>	<ul style="list-style-type: none"> <li>• Anthropische KI Schlüssel</li> </ul>	<a href="#">Anthropin</a>	<a href="#">Arbeitsblatt</a> <a href="#">Eule</a>
<a href="#">IBM Das ist nicht möglich. Authentischer</a>	<ul style="list-style-type: none"> <li>• <a href="#">IBM watsonx.ai</a> <a href="#">LLM-Auswahl</a></li> <li>• <a href="#">IBM watsonx.ai</a> <a href="#">Einbettungsmodell</a> <a href="#">Anschluss</a></li> </ul>	<ul style="list-style-type: none"> <li>• IBM wasonx.ai API Schlüssel</li> <li>• Projekt oder Raum Verbindung</li> </ul>	<a href="#">IBM</a>	<a href="#">Arbeitsblatt</a> <a href="#">Eule</a>
<a href="#">Tiefsee Authentischer</a>	<ul style="list-style-type: none"> <li>• <a href="#">Tiefsee LLM</a> <a href="#">Auswählen</a></li> </ul>	<ul style="list-style-type: none"> <li>• DeepSeek API Schlüssel</li> <li>• Basis-URL (fakultativ)</li> </ul>	<a href="#">Tiefsee</a>	<a href="#">Arbeitsblatt</a> <a href="#">Eule</a>
<a href="#">Google AI Studio Authentischer</a>	<ul style="list-style-type: none"> <li>• <a href="#">Vertex AI Stecker</a></li> </ul>	<ul style="list-style-type: none"> <li>• Google Clouds Projekt-ID</li> <li>• Google Clouds Standort</li> </ul>	<a href="#">Vertex ai</a>	<a href="#">Arbeitsblatt</a> <a href="#">Eule</a>
<a href="#">Datenbrände Arbeitsraum Anschluss</a>	<ul style="list-style-type: none"> <li>• <a href="#">Databricks LLM</a> <a href="#">Auswählen</a></li> <li>• <a href="#">Datenbrände</a> <a href="#">Einbettungen Modell</a> <a href="#">Auswählen</a></li> </ul>	<ul style="list-style-type: none"> <li>• Databricks Arbeitsraum URL</li> <li>• Persönliches Zugang zu den</li> </ul>	<a href="#">Datenbrände</a>	<a href="#">Arbeitsblatt</a> <a href="#">Eule</a>

# Retrieval-Augmented Generation (RAG) in KNIME

Große Sprachmodelle (LLMs) sind leistungsstark, aber begrenzt. Sie haben keinen Zugang zu privaten Dokumenten oder Echtzeit-Informationen und können mit neuem Wissen nicht einfach aktualisiert werden. Retrieval-Augmented Generation (RAG) löst dies, indem externe Informationen in die auf Anfrage Zeit.

Um eine RAG-Pipeline in KNIME aufzubauen:

1. [Verbinden mit Modellanbieter](#page20)
2. [Erstellen eines Vektorspeichers](#page21)
3. [Kontext abrufen](#page22)
- [Antwort generieren](#page22)

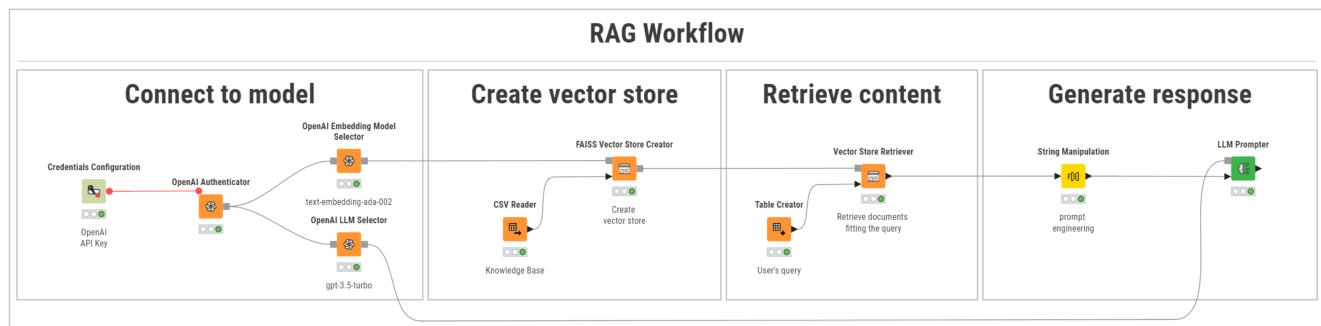


Abbildung 18. Vier Phasen einer RAG-Pipeline in KNIME: an ein Modell anschließen, einen Vektorspeicher erstellen, einen relevanten Inhalt abrufen und eine geerdete Antwort generieren.

## Verbinden mit Modellanbieter

Bevor Sie Dokumente abrufen oder Antworten generieren, verbinden Sie sich mit einem Sprachmodellanbieter und wählen Sie ein Einbettungsmodell.

Verwenden Sie die folgenden Knoten:

- [Konfigurieren von Anmeldeinformationen](#)

Speichern Sie sicher Ihren API-Schlüssel oder Token.

- [Authenticator Node \(z. \[OpenAI Authentication\]\(#\)\)](#)

Authenticates Zugriff auf Ihre Einbettung und LLM-Anbieter.

- Modellauswahl (z.B., [OpenAI Embedding Modellauswahl](#))

Wählen Sie das Einbettungsmodell, das verwendet wird, um Ihre Dokumente zu vektorisieren

- LLM Selector (z.B., [OpenAI LLM Selector](#))

Später verwendet, um die endgültige Antwort aus dem abgerufenen Kontext zu erzeugen.

[Für eine vollständige Liste der unterstützten Modelle, siehe die](#page18)

## Vector Store erstellen

Ein Vektorspeicher ist ein Index von Einbettungen, die semantische Suche durch Mapping ermöglicht

Dokumente zu numerischen Vektoren. KNIME unterstützt zwei Backends:

FAISS	Flacher Datei-basierter Index optimiert für schnelles, lokales Abrufen.
Chrom	JSON + SQLite Format mit Unterstützung von Metadaten und Dokumentensammlungen.

Um den Vektorspeicher zu bauen, verwenden Sie einen der folgenden Knoten:

- [FAISS Vector Store Creator](#)

Erzeugt einen FAISS-Index mit dem ausgewählten Einbettungsmodell. Sie können auch eine Säule mit vorkomputierten Einbettungen statt in-node zu erzeugen.

- [Chroma Vector Store Creator](#)

Erstellt einen Chroma-Store mit optionalen Metadaten für erweiterte Filterung oder Gruppierung. Unterstützt sowohl inline als auch vorkomputierte Einbettungen.

Um einen Vektorspeicher über Workflows wiederzuverwenden:

- Speichern Sie es mit dem [Modellschreiber](#).
- Nachladen mit [Modell Reader](#).

Chroma hat sein Datenlayout aktualisiert. Ältere Vektorspeicher können vorübergehend benötigen

[Kopien zum Lastzeitpunkt. Verwenden Sie die](#) [Vector Store Datenextraktion](#)

Knoten zu migrieren

bestehende Daten in einen neuen Speicher.

Zurück zum Inhalt

Sobald Ihre Dokumente in einem Vektorspeicher indexiert sind, können Sie die relevanten Einträge abrufen für eine Benutzeranfrage.

Im obigen Beispiel-Workflow wird der Vektorspeicher mit Hilfe der

[FAISS Vector](#)

[Store Creator](#) Knoten und direkt an den Abrufer übergeben.

Wenn Sie stattdessen einen gespeicherten Vektorspeicher (z.B. von der Festplatte oder einem anderen Workflow) wiederverwenden möchten, verwenden einer dieser Knoten, um es zu laden:

- [FAISS Vector Store Reader](#)

- [Chroma Vector Store Reader](#)

Dann folgen Sie diesen Schritten:

#### ANH ANG Eingabe der Benutzeranfrage

Verwenden Sie die [Tabelle Schöpfer](#) die Frage des Benutzers zu simulieren. Dies erzeugt einen String Spalte, die die Eingabeabfrage darstellt. Jeder Knoten, der einen String ausgibt, ist hier gültig.

#### 2. Retrieve relevanter Kontext

[Die Abfrage mit der](#) [Vector Store Retriever](#) Knoten. Die Abfrage wird mit der gleiches Modell wie der Vektorspeicher und liefert die semantisch ähnliche Ergebnisse.

Die Ausgabe liefert die kontextuellen Dokumente, die die letzte Antwort der LLM auslösen werden.

Antwort generieren

Nach dem Abrufen des Kontexts, bereiten Sie die endgültige Aufforderung, indem die Ergebnisse in eine einzelne String.

- Verwendung [String Manipulation](#) den Kontext zu formatieren:

```
JOINSEP("\n", Spalte("Answer"))
```

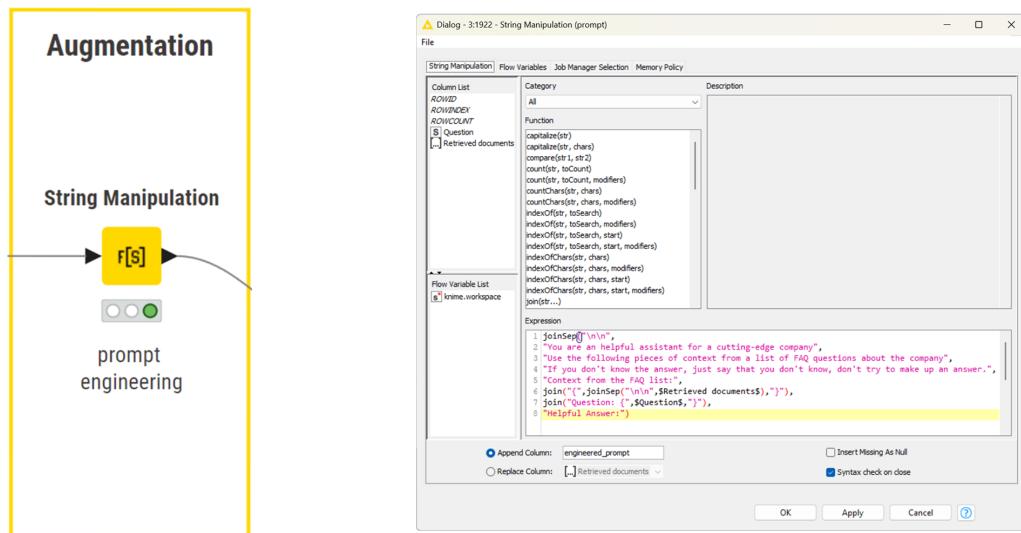


Abbildung 19. Die Eingabeaufforderung, die durch die Kombination der Abfrage und des abgerufenen Kontexts erstellt wird.

- Senden Sie die vollständige Aufforderung (Frage + abgerufener Kontext) an die LLM mit:

[LLM Promoter](#)

Die endgültige Antwort wird in einer Spalte gespeichert, die benannt ist Antwort .

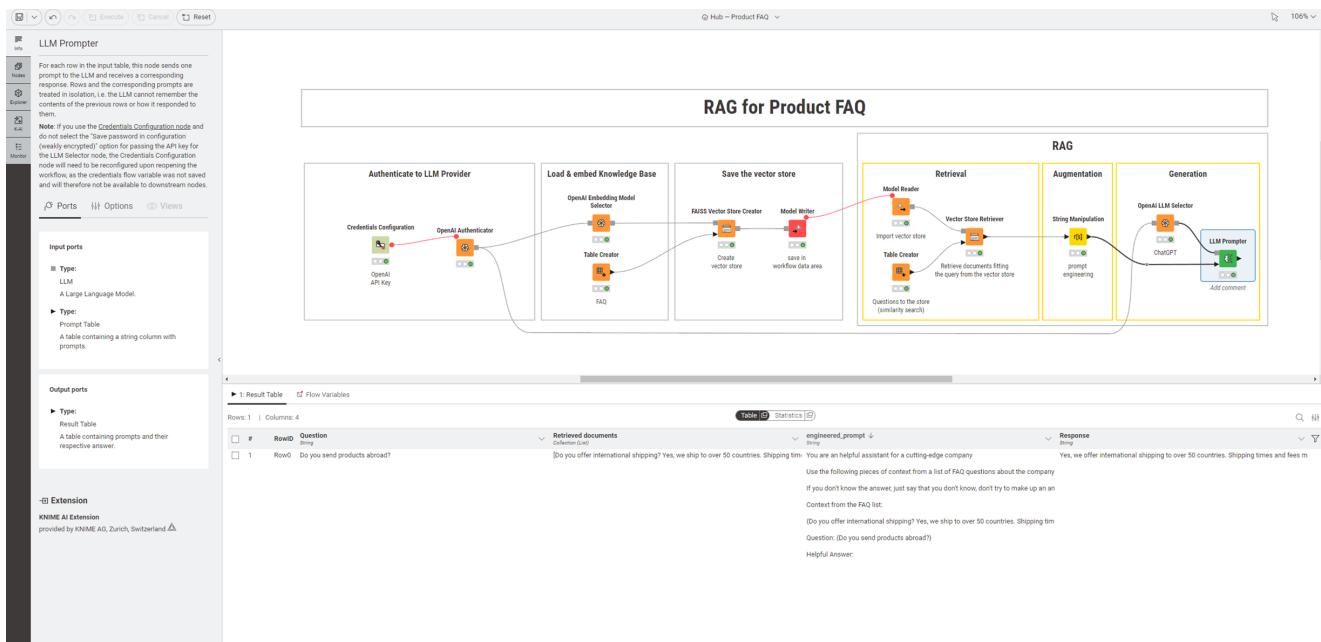


Abbildung 20. Ausgabe des RAG-Prozesses: die vom LLM generierte geerdete Antwort.

## Beispiel: Produkt FAQ Assistentin der RAG

Sie möchten einen Assistenten aufbauen, der Fragen zu den Produkten Ihres Unternehmens beantworten kann und Dienstleistungen. Um dies zu tun, entscheiden Sie sich für eine Retrieval-Augmented Generation (RAG) Architektur, mit einer Datei, die häufig gestellte Fragen (FAQs) als Ihr Wissen enthält

Basis.

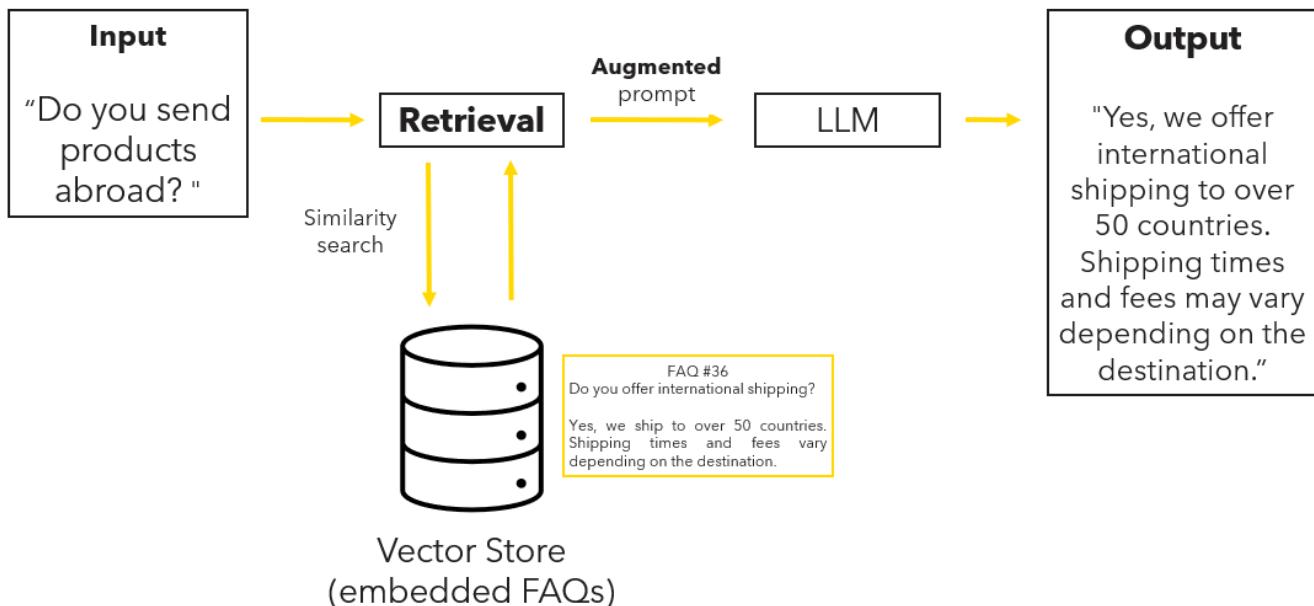


Abbildung 21. Einen Überblick über eine RAG-Pipeline

Beispiel FAQ-Daten (CSV)

So sieht die Datei mit FAQ aus:

ID	FAQ
1	Was ist die Rückgabepolitik? Sie haben das Recht, ein Produkt innerhalb von 30 Tagen zurückzusenden
2.	Wie kann ich mein Passwort zurücksetzen? Sie können Ihr Passwort zurücksetzen, indem Sie auf 'Forgot Password' auf der Anmeldeseite und folgen den Anweisungen.
3	Bieten Sie internationalen Versand an? Ja, wir versenden in über 50 Länder. Versand Zeiten und Gebühren variieren je nach Ziel.

### Workflow: FAQ Product Assistant

Dieser Workflow implementiert Retrieval-Augmented Generation (RAG) auf einer Produkt-FAQ-Datei. Es umfasst drei Hauptschritte: Authentifizierung, Vektorspeicher-Erstellung und retrieval-augmented Generation.

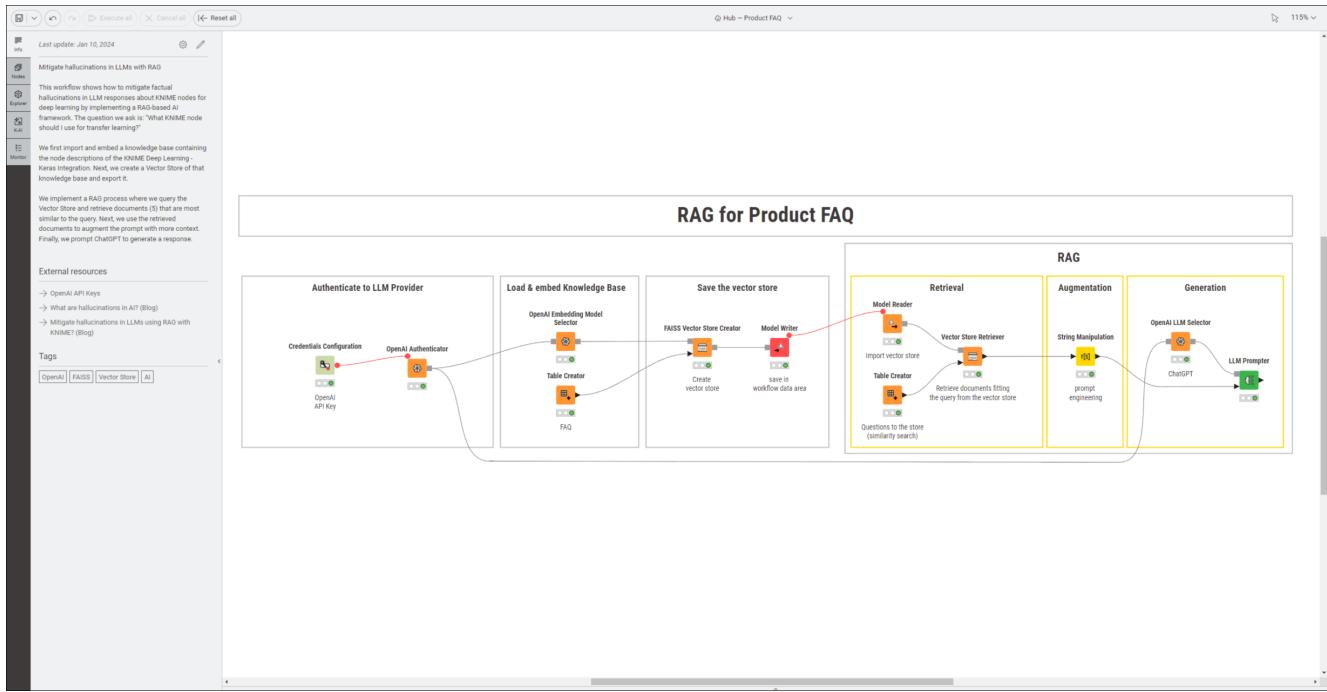


Abbildung 22. Arbeitsablauf die eine RAG-Architektur verwendet, um FAQs zu beantworten

## ANHANG Authenticate

### • Geben Sie API-Anmeldeinformationen

Verwenden Sie die [Konfigurieren von Anmeldeinformationen](#) node to store the OpenAI API key.

### • Authenticate

Die [OpenAI Authentication](#) node authentifiziert die Verbindung zu OpenAI.

## 2. Vector Store erstellen

### • Wählen Sie Einbettungsmodell

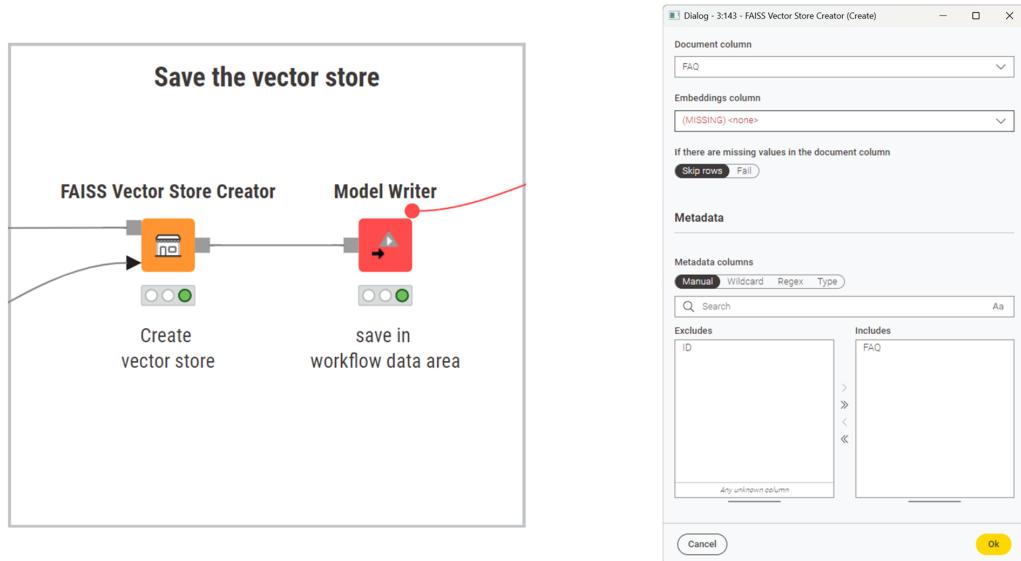
Die [OpenAI Embedding Modellauswahl](#) node wählt das Einbettungsmodell aus (text-Einbettung-3-klein).

### • Lesen Sie Daten

[CSV Reader](#) node lädt die FAQ-Datei.

### • Vector Store erstellen

Die [FAISS Vector Store Creator](#) erstellt Einbettungen und speichert sie in einem Vector Store

**Abbildung 23.** Konfigurationsdialog der**[FAISS Vector Store Creator](#)**

. In diesem Beispiel der Knoten

erzeugt Einbettungen intern, da keine vorgefertigten Einbettungen vorgesehen sind. Die KI

Erweiterung beinhaltet auch einen dedizierten Knoten für die Einbettung Generation: das Text Embedder (siehe  
[früherer Abschnitt](#page14))

### 3. Retrieval Augmented Generation (RAG)

#### Rücknahme

- **Last Vector Store**

Die [Modell Reader](#) node lädt den gespeicherten Vector Store von disk. Die [FAISS Vector Store Reader](#) node bringt den Speicher in den Speicher.

- **Benutzeranfrage bereitstellen**

Die [Tabelle Schöpfer](#) node simuliert eine Benutzeranfrage.

- **Abfrageeinbettung generieren**

Die [OpenAI Embeddings Connector](#) node erzeugt eine Einbettung für die Benutzeranfrage mit dem gleichen Einbettungsmodell.

- **Ähnliche Einträge abrufen**

Die [Vector Store Retriever](#) node vergleicht die Abfrageeinbettung gegen die gespeicherte Einbettungen, um die ähnlichsten FAQ-Einträge abzurufen. In diesem Beispiel ist die Anzahl der abgerufene Ergebnisse werden aufgrund des kleinen Datensatzes auf 1 gesetzt. In realen Anwendungsfällen mehrere Ergebnisse können die Erdung verbessern, indem das Modell mehr Kontext bietet.

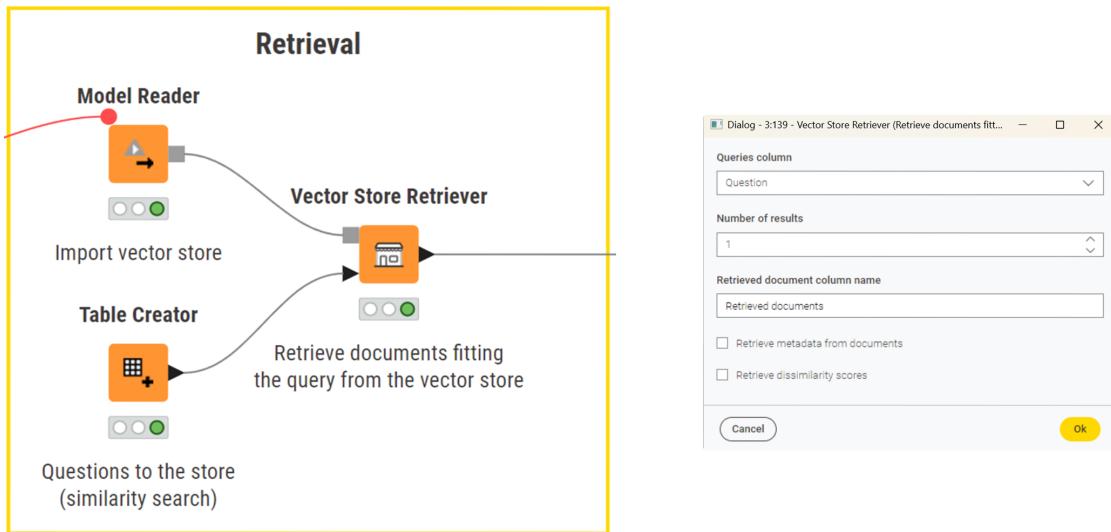


Abbildung 24. Konfigurationsdialog des Vector Store Retriever-Knotens.

## Erweiterung

### • Kontext

Die [String Manipulation](#) node verbindet mehrere abgerufene FAQ Antworten in eine einzelne String mit:

```
JOINSEP("\n", Spalte("Answer"))
```

Dies schafft einen kombinierten Kontextblock für das Sprachmodell.

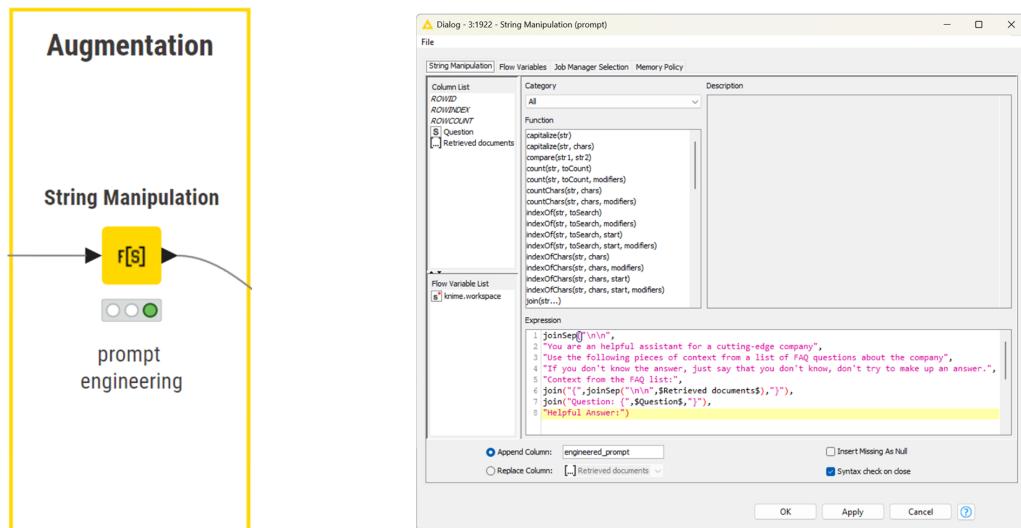


Abbildung 25. Prompt mit Hilfe von

[String Manipulation](#)

Knoten.

## Generation

### • Prompt an Modell senden

Die [LLM Prompter](#) node sendet sowohl die Benutzerfrage als auch den abgerufenen FAQ-Kontext an das Sprachmodell.

### • Ausgangsanswort

Die Antwort des Modells wird in eine neue Spalte geschrieben Antwort.

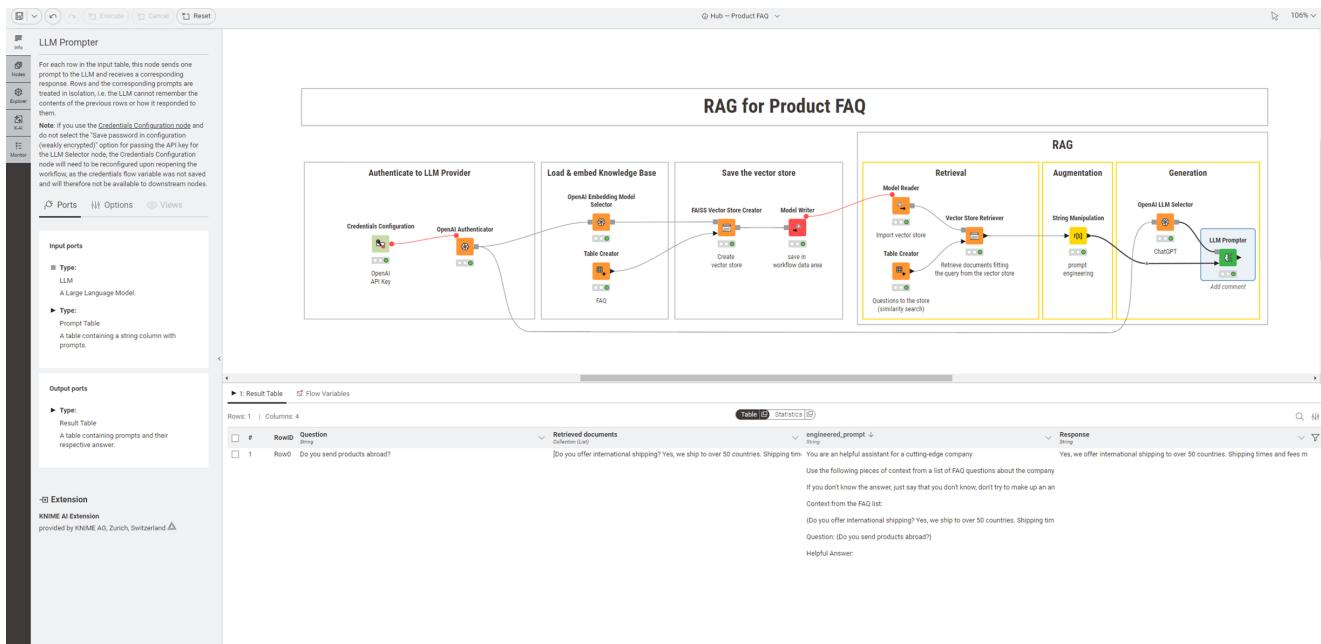


Abbildung 26. LLM Ausgabevorschau in der Antwortspalte

## Agentische KI in KNIME

Eine **Mittel** in KNIME ist ein Workflow, der mit einem großen Sprachmodell (LLM) Aufgaben löst. Es kann die Anfrage eines Benutzers lesen, Schritt für Schritt denken und andere Workflows ausführen, genannt die Aufgabe zu erledigen.

Werkzeuge, bis

[Um einen Agenten-Workflow zu erstellen](#)    [KNIME Analytics Plattform](#) , folgen Sie diesen Schritten:

- [1. \*\*ANH\*\* Kontaktieren Sie den LLM-Anbieter](#page30)
- [2. \*\*ANG\*\* Zugriffstools](#page31)
- [3. \*\*ANG\*\* Wirkstoff](#page31)
- [4. \*\*ANG\*\* Dateneingabe bereitstellen \(durch Knime\)](#page34)

L.MDF  
Version 2024.01  
3.5.21

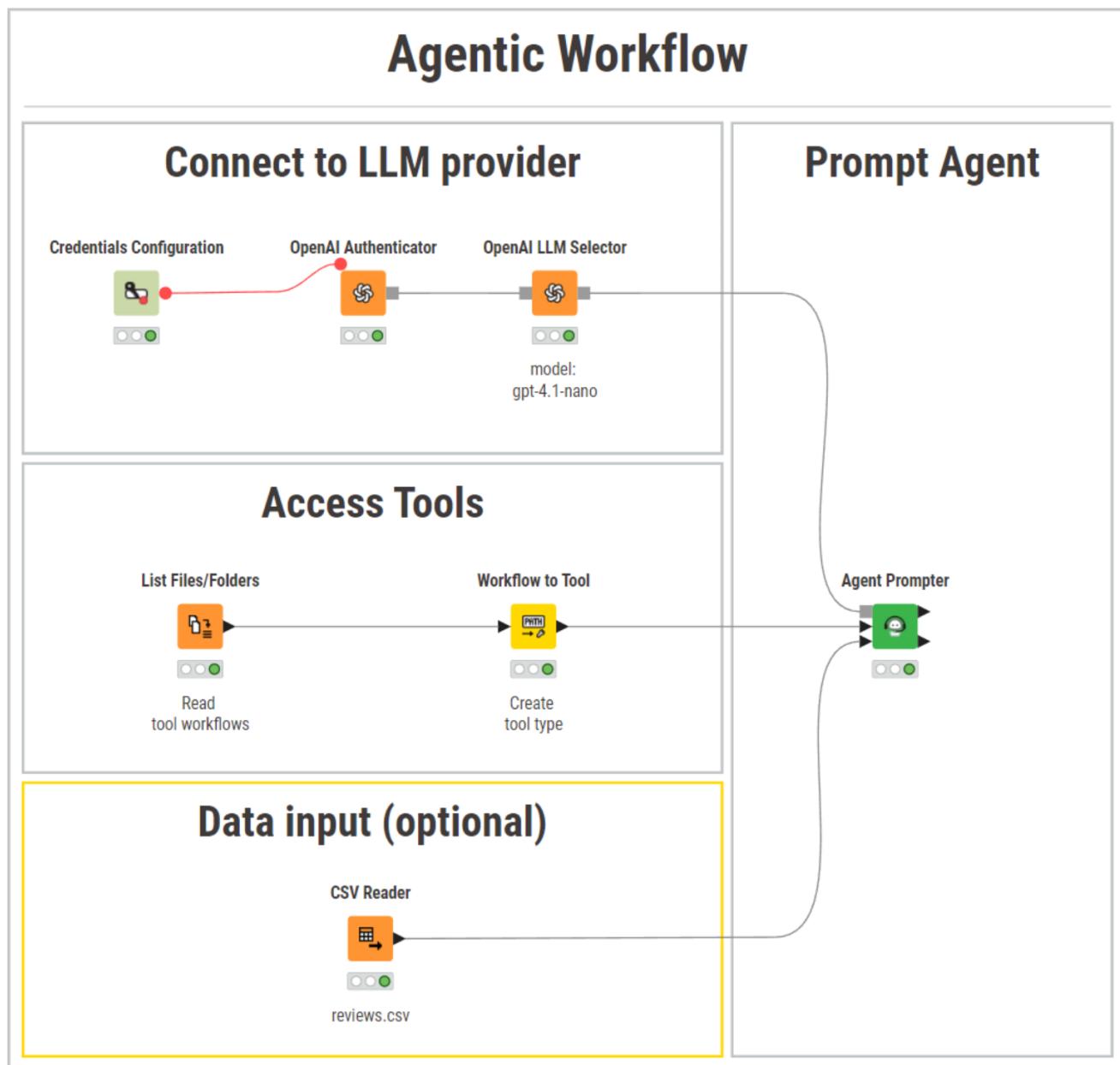


Abbildung 27. Die Architektur eines agentischen Workflows in KNIME zeigt die wichtigsten Schritte: verbinden auf einen LLM-Anbieter, auf die verfügbaren Tools zugreifen und den Agenten veranlassen, mit Gründen zu beginnen. Optional können Daten als Eingabe bereitgestellt werden.

### Anschluss an LLM Anbieter

Um Ihren Agenten zu verdanken, verbinden Sie es mit einem LLM.

KNIME bietet dazu dedizierte Knoten:

- Authenticator-Knoten (z. [OpenAI Authentication](#) )

Speichert Ihren API-Schlüssel oder Token, um auf den gewählten Sprachmodellanbieter zuzugreifen.

- [LLM Selector-Knoten](#) (z.

- [OpenAI LLM Selector](#)

:

Wählen Sie das Sprachmodell aus (z.B.  
grundierungsprozess.

GPT-4.1-nano

) die den Agenten

Für unterstützte Modelle und Konfigurationstipps siehe

<a href="#page18" style="color: #ff6600; text-decoration: underline;">Hier

## Zugriff auf Tools

Werkzeuge sind rufbare Workflows, die einzelne Subtasks ausführen, wie eine Frage zu beantworten,  
Überprüfung einer Bedingung oder Transformation eines Datensatzes.

Um dem Agenten Werkzeuge zur Verfügung zu stellen, verwenden Sie:

- [Dateien/Folders auflisten](#) Knotenpunkt

Scannen Sie den Ordner, in dem Ihre Tool-Workflows gespeichert sind (z.B. .knwf Dateien).

- [Workflow zum Werkzeug](#) Knotenpunkt

Konvertiert jeden Workflow in ein Werkzeug, über das der Agent ausrichten kann.

Dieser Schritt stellt sicher, dass der Agent weiß:

- Was jedes Werkzeug tut (über seine Beschreibung).
- Welche Eingaben es benötigt (Parameter oder Datensätze).
- Was gibt es aus (Botschaften oder Tabellen).

Zur Führung des Aufbaus eines Werkzeug-Workflows siehe Abschnitt

[Einen Workflow in ein](#)

[Werkzeug](#).

## Prompt Agent

Nach dem Verbinden eines Sprachmodells und Registrierungstools können Sie den Agenten anfordern, zu beginnen  
zu urteilen. KNIME bietet hierfür zwei Knoten:

- [Agent Prompter](#)
- [Agent Chat anzeigen](#)

Jeder unterstützt verschiedene Anwendungsfälle, wie unten beschrieben.

## Agent Promter Knoten

Verwenden Sie die [Agent Promter](#) Knoten, um die Argumentationsschleife des Agenten innerhalb eines Workflows zu starten.

Dieser Knoten ist am besten für automatisierte Ausführung oder Debugging einzelner Aufforderungen in Workflows, ohne Benutzerinteraktion.

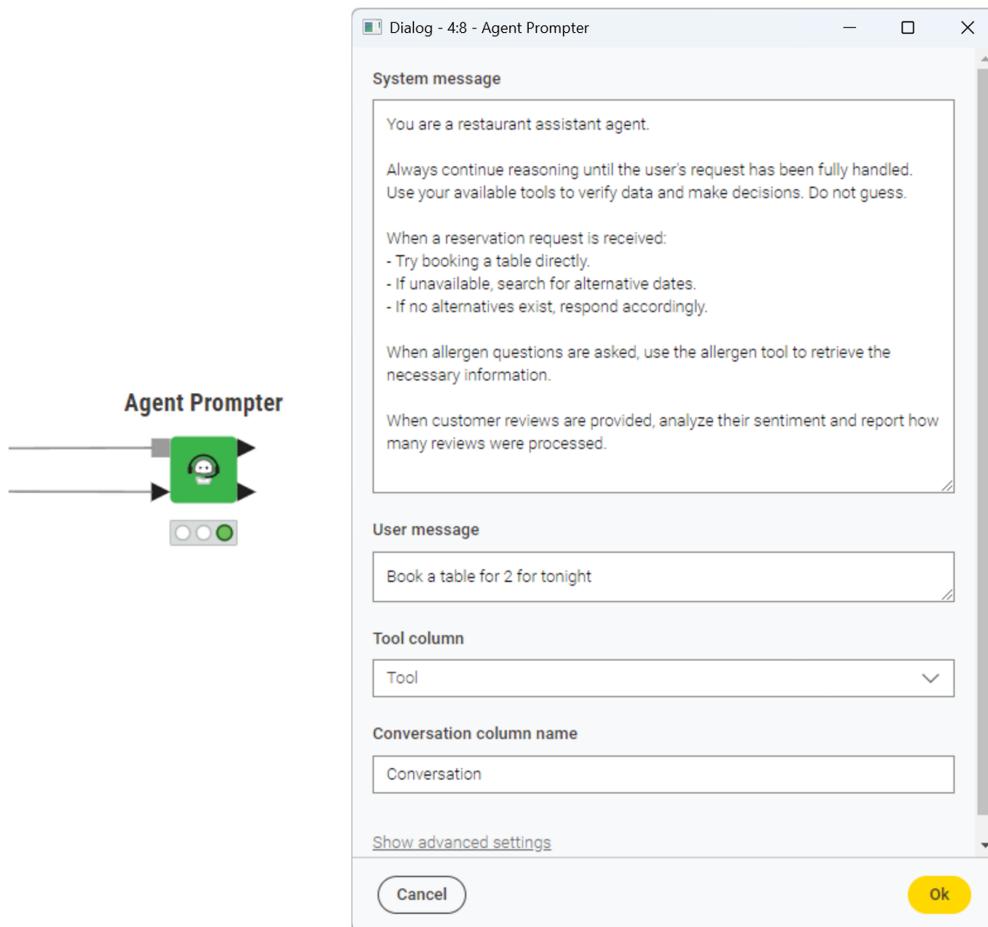


Abbildung 28. [Agent Promter](#) Knotenkonfigurationsdialog.

Der Knoten nimmt die folgenden Eingänge:

- **Systemnachricht** : definiert die Rolle, das Verhalten oder die Regeln des Agenten. Beispiel : "Du bist ein Support Agent, der Produktfragen beantwortet."
- **Benutzernachricht** : die eigentliche Aufgabe oder Frage zu lösen. Beispiel : "Was ist die Garantie? für das Produkt X?"
- **Artikelliste** : die Menge der verfügbaren Tools, die mit der [Workflow zum Werkzeug](#) Knoten.

## Agent Chat View node

Um Ihren Agenten interaktiv zu machen, verwenden Sie die [Agent Chat anzeigen](#) Knoten. Dies öffnet einen Live-Chat

Schnittstelle, wo Benutzer mit dem Agenten sprechen können, Fragen stellen und Antworten in real erhalten Zeit.

Verwenden Sie diesen Knoten, wenn Sie eine Assistent-Stil-Schnittstelle für Endbenutzer bauen möchten.

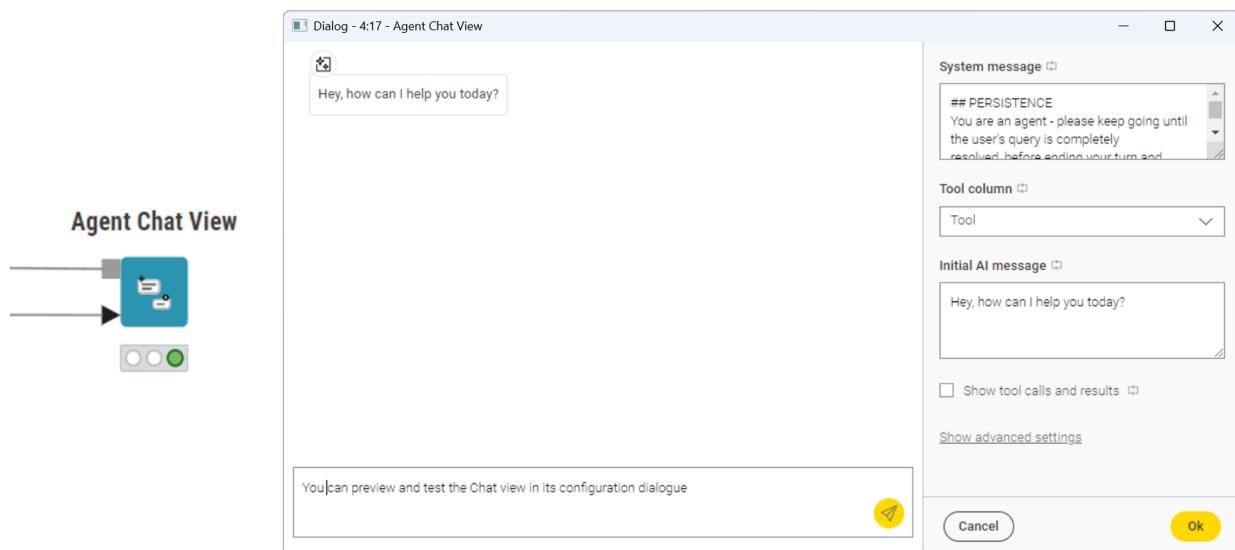


Abbildung 29. Der Konfigurationsdialog der

[Agent Chat anzeigen](#)

Knotenpunkt

Dieser Knoten:

- Systemnachricht : definiert die Rolle, Zweck und Verhalten des Agenten für das Gespräch.
- Werkzeugsäule : eine Spalte mit der Liste der verfügbaren Tools (aus der [Workflow zum Werkzeug](#) Knoten)
- Erste AI-Nachricht (optional) : eine Gruß- oder Eröffnungsnachricht vor dem Benutzer sendet alles Beispiel : „Hey, wie kann ich dir heute helfen? „

Während der Ausführung können Benutzer Fragen oder Anfragen eingeben. Die Agentengründe durch die anfordern, Werkzeuge verwenden, wenn nötig, und reagiert in Echtzeit.

Sie können auch wählen, was Benutzer sehen:

- Wenn du tickst Werkzeuganrufe und Ergebnisse anzeigen die Schnittstelle zeigt das vollständige Gespräch an. Das beinhaltet die interne Begründung, die Werkzeugnutzung und die Antworten.
- Wenn Sie es unbemerkt lassen, sieht der Benutzer nur die letzten Antworten des Agenten, so dass die Interaktion fühlen sich eher wie ein typischer Assistent Chat.

Sie können die [Agent Chat anzeigen](#) in [Komponente](#), dann setzen Sie es [KNIME Hubraum](#) bis Ihren Agenten als interaktiver Service für Endbenutzer zur Verfügung stellen.

Wenn ein Werkzeug-Workflow Knoten umfasst, die Ansichten generieren (z. [Scatter Plot](#) ), Agent kann das Tool anrufen und die resultierende Ansicht wird im Chat angezeigt Schnittstelle. Diese Funktionalität ist derzeit nicht kompatibel mit KNIME Business Hubversionen 1.15 und früher.

## Eingangsdaten

Einige Tools funktionieren auf Datentabellen. z.B. Filtern von Datensätzen, Auswertungen oder Zusammenfassungen generieren. In diesen Fällen muss der Agent Daten an das Werkzeug übergeben und optional das Ergebnis nach der Verarbeitung abrufen.

Standardmäßig, die [Agent Prompter](#) und [Agent Chat anzeigen](#) Knoten enthalten keine Datenports.

Um den Datenaustausch zu ermöglichen:

ANHANG Klicken Sie auf den Knoten im Workflow-Editor.

2. Wählen Sie das Plus-Symbol aus, das angezeigt wird.

3. Wählen Input Port hinzufügen oder Ausgabeport hinzufügen, und wählen Sie einen Datentabellentyp.

Die [Agent Prompter](#) node erlaubt sowohl Eingabe- als auch Ausgabeports. Die [Agent Chat anzeigen](#) Stützen nur eingeben.

Beachten Sie, dass der Agent selbst nie die Tabelle direkt inspiert. Stattdessen nennt es ein Werkzeug, dass verarbeitet die Daten und gibt ein Ergebnis zurück. Der Agent nutzt das Ergebnis dann weiter zu urteilen.

Für Details, wie Daten durch Werkzeuge fließen, siehe den dedizierten Abschnitt auf der

<a href="#page38" style="color: #ff6600; text-decoration: underline;">.

## Wie man Tools erstellt

Für die Verwendung eines Arbeitsablaufs als Werkzeug muss der Arbeitsablauf so strukturiert werden, daß Agent kann verstehen und ausführen. Dieser Abschnitt zeigt, wie man einen KNIME-Workflow vorbereitet der Agent kann daran denken, in Eingaben passieren und Ergebnisse abrufen.

Jedes Tool interagiert mit dem Mittel durch zwei Schichten:

ANH  
ANG

[Kommunikationsschicht](#page35)

Zeigt den Austausch von Nachrichten zwischen dem Agenten und dem Werkzeug. Es ermöglicht

Agent zu entscheiden, welches Werkzeug zu verwenden, warum und wie, basierend auf Benutzer-Intention und die Aufgabe bei Hand.

[Datenschicht](#page38)  
2.

Verwaltet den tatsächlichen Datenfluss. Obwohl der Agent nicht direkt Datentabellen anzeigen kann, können Tools aktivieren, die auf Daten im Hintergrund arbeiten und Ergebnisse zurückgeben.

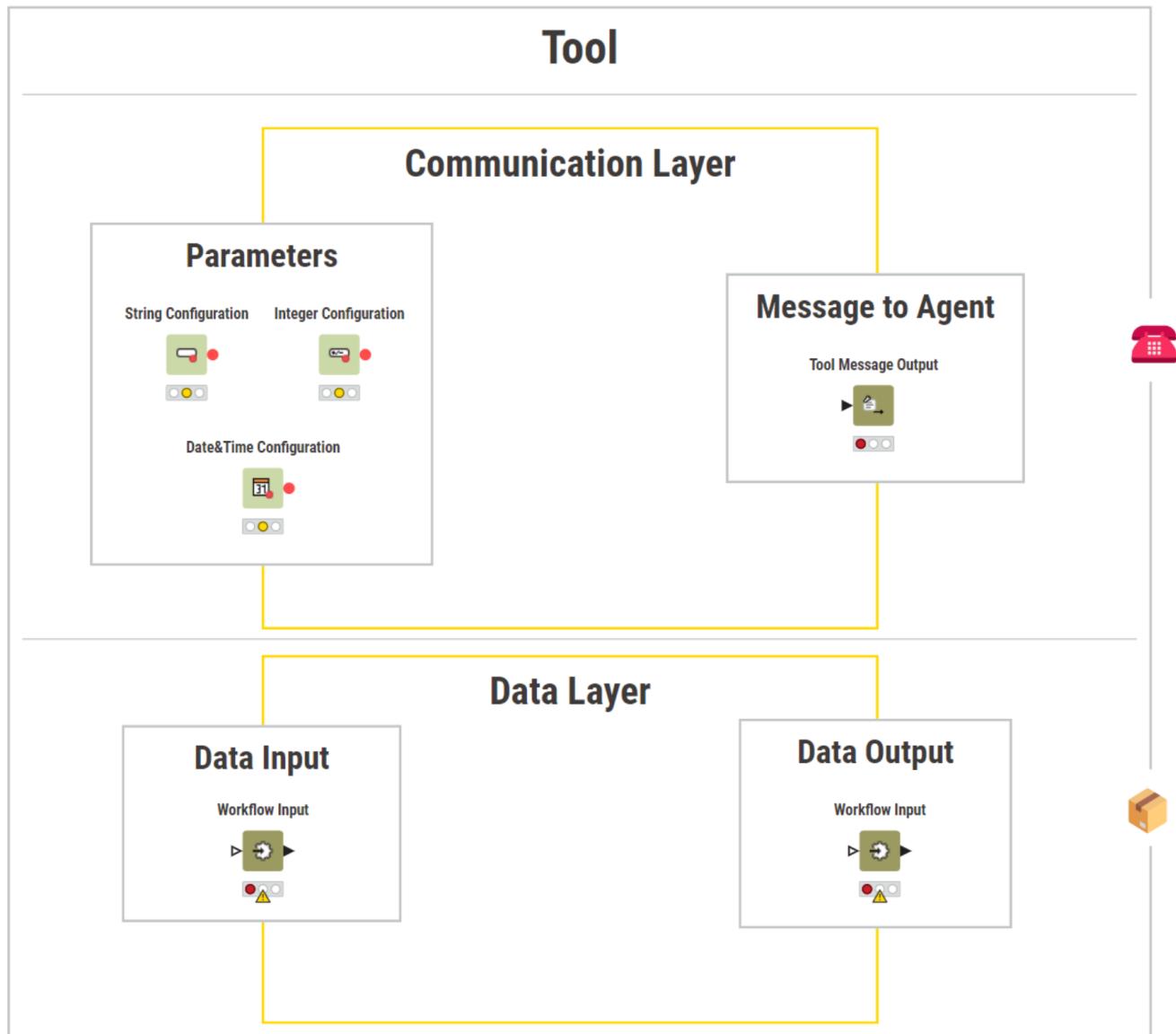


Abbildung 30. Aufbau eines Tool-Workflows in der KNIME Analytics Platform

Kommunikationsschicht: Führen Sie die Argumentation des Agenten

Werkzeugverhalten für den Agenten beschreiben

Jedes Werkzeug muss eine klare Beschreibung im Info-Bereich des Workflows enthalten. Der Agent liest

diese Beschreibung, um festzustellen, wann und wie das Werkzeug verwendet werden soll.

Die Beschreibung sollte beinhalten:

- Was das Werkzeug tut
- Welche Inputs es erwartet
- Was es zurückgibt
- Die Arten von Anfragen, die es handhaben kann

Eine genaue Beschreibung verbessert die Fähigkeit des Agenten, über welches Werkzeug zu verwenden in Antwort auf die Anfrage eines Benutzers.

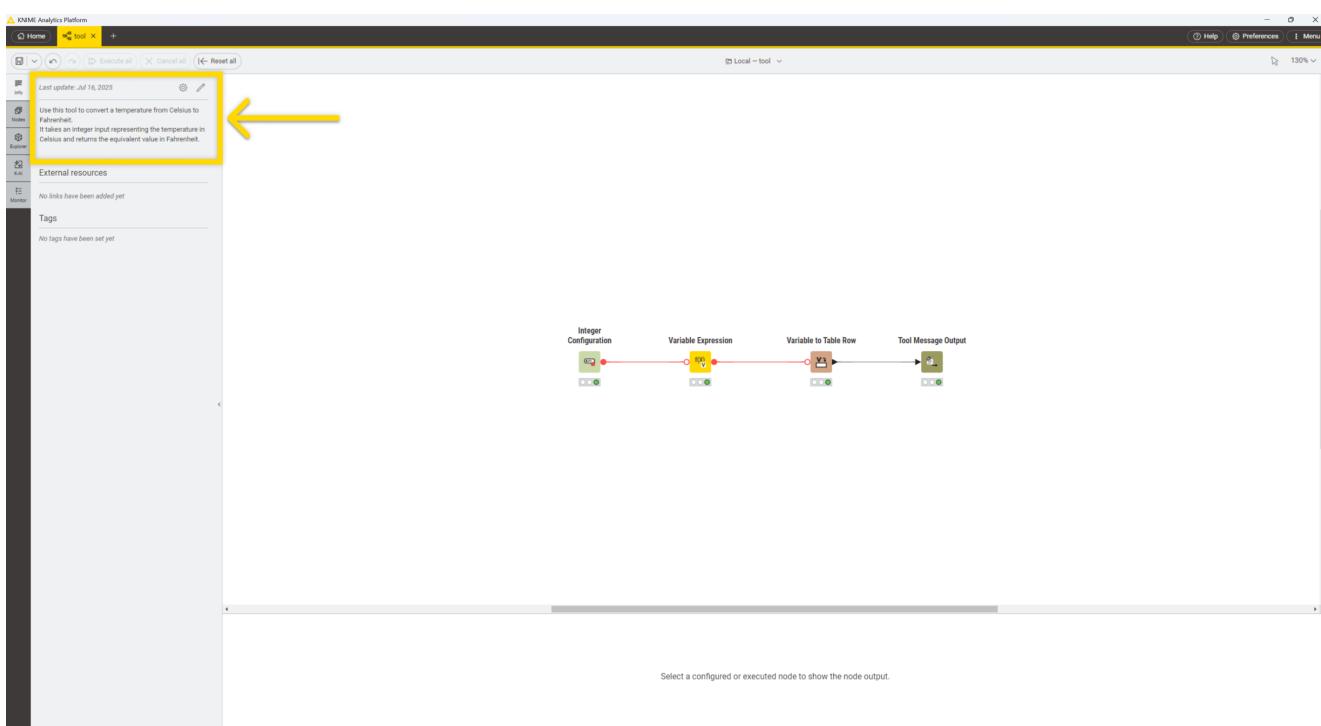


Abbildung 31. Das Werkzeugbeschreibungsfeld in der KNIME Analytics Platform. Dies wird vom Mittel zur verstehen, wann das Tool aufgerufen wird.

#### Return Results to the Agent

Um dem Agenten eine aussagerelevante Rückmeldung zu senden, verwenden Sie die

#### [Tool-Nachrichtenausgang](#)

Knoten.

- Fügen Sie diesen Knoten hinzu, wenn das Tool eine Nachricht zurückgeben muss, die der Agent lesen kann und Grund mit.
- Lassen Sie es aus, wenn das Tool nur strukturierte Daten zurückgibt.

Der Knoten liest die erste Zelle (Reihe 1, Spalte 1) seiner Eingabetabelle und sendet sie als Ergebnis Nachricht an den Agenten.

Verwenden Sie es, um zurückzukehren:

- Zusammenfassungen der verarbeiteten Daten.
- Schlüsselergebnisse oder Bestätigungen.
- Vorläufige Argumentationsschritte.



Abbildung 32: Eine Tabelle an die [Tool-Nachrichtenausgang](#) Knoten. Nur die erste Zelle (erste Zeile, erste Spalte) wird als Ausgabenachricht verwendet.

Lassen Sie die Parameter des Agenten setzen

KNIME Konfigurationsknoten (z. [String Konfiguration](#), [Integer Konfiguration](#)) bis einstellbare Parameter definieren.

Jeder Parameter sollte haben:

- Ein klarer Name (als Variable verwendet)
- Eine kurze Beschreibung, die erklärt, was es kontrolliert

Das Mittel verwendet diese Metadaten, um Parameter dynamisch zuzuordnen.

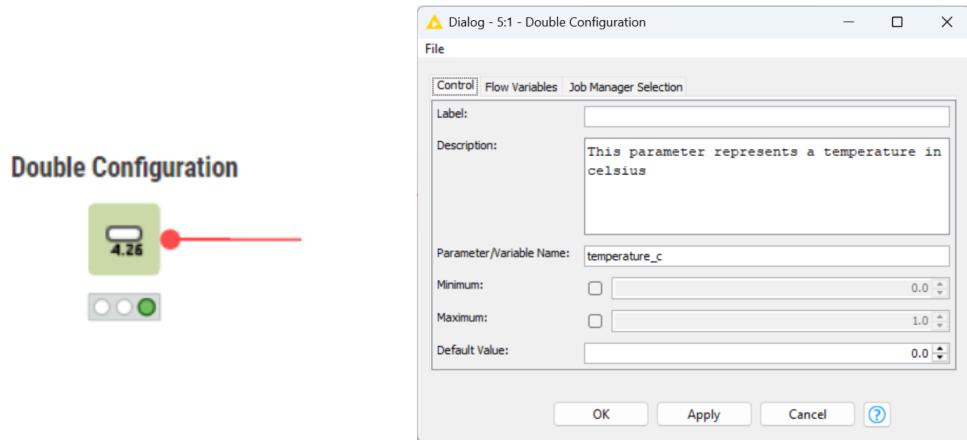


Abbildung 33. Das Konfigurationsfenster für einen Parameter, mit Feldern für Name und Beschreibung bis führen Sie den Agenten.

Datenschicht: automatisieren Datenfluss

Definieren Sie, welche Daten das Tool verwendet und zurückgibt

Verwenden Sie die [Workflow Input](#) Node, um die Struktur der Dateneingabe in einem Werkzeug anzugeben.

Fügen Sie im Knotenkonfigurationsdialog eine klare Beschreibung der Eingabetabelle hinzu, dass das Tool erwartet. Achten Sie darauf, Spaltennamen und Datentypen einzubeziehen.

Verwenden Sie die [Workflow-Ausgang](#) die vom Werkzeug erzeugten Daten zu beschreiben.

Die Bereitstellung einer Beschreibung der Ausgabetafel hilft dem Agenten zu verstehen, wie man das Ergebnis verwendet.

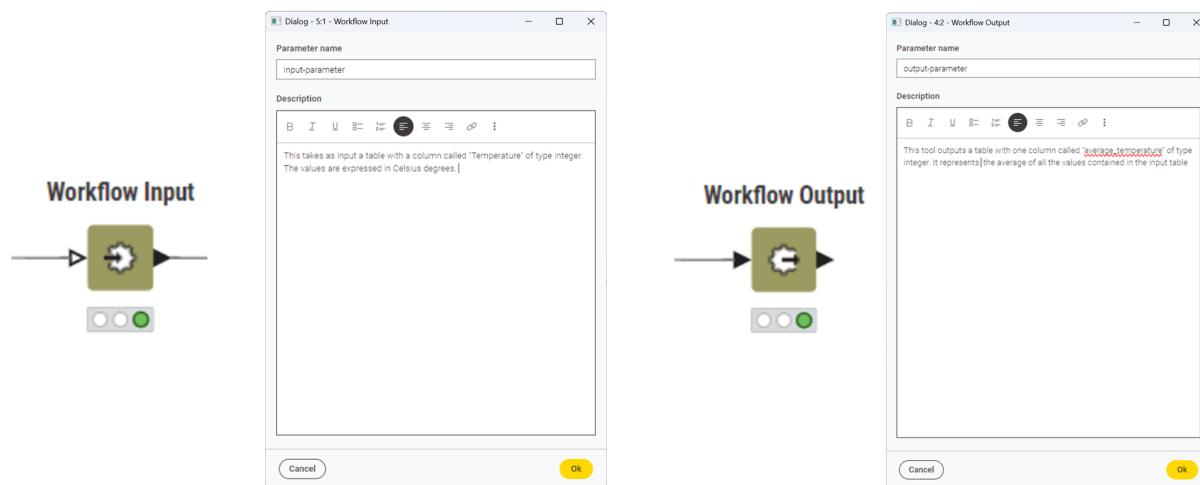


Abbildung 34. Die Workflow Input- und Output-Knoten enthalten Beschreibungen der von das Werkzeug

Das Mittel **nicht** Zugriff auf Rohdaten. Stattdessen kann es Werkzeuge nennen, die Daten verarbeiten Verarbeitung und Rückgabe von Zusammenfassungen oder strukturierten Ergebnissen.

## Werkzeuge für den Agenten entdecken

Einmal Werkzeug-Workflows sind bereit, Sie müssen sie registrieren, so dass der Agent finden und verwenden kann sie.

Dazu:

ANHANG Legen Sie alle Tool-Workflows in einem einzigen Ordner.

2. In demselben Verzeichnis erstellen Sie einen neuen Workflow, um als Werkzeugregistrierung zu fungieren.
3. Verwenden Sie die [Dateien/Folders auflisten](#) Knoten, um alle Workflows im Ordner zu lesen.

[L 347 vom 20.12.2013, S. 1\)](#). Geben Sie die Liste an die [Workflow zum Werkzeug](#) Knoten, um jeden in ein rufbares Tool zu konvertieren.

Der Ausgang der [Workflow zum Werkzeug](#) Knoten enthält Metadaten für jedes Tool, so dass Sie überprüfen sie sind richtig eingerichtet:

- zeigt, ob eine Beschreibung vorliegt.
- zeigt, wie viele Parameter definiert sind.
- zeigt, wie viele Dateneingänge und -ausgänge verfügbar sind.

Damit ist es einfach, auf einen Blick zu prüfen, ob Werkzeuge komplett und gebrauchsfertig sind.

Tool Tool			
Calculator	1	1	
Country_populations	1	1	
No_params		1	1
WebpageRetriever	2	1	1
Workflow Builder	1	1	1
complete tool with ...	2	2	3
select_column	1	1	1

Abbildung 35. Ausgabe der [Workflow zum Werkzeug](#) die Metadaten für jedes Tool anzeigen.

Checkliste: Was Sie brauchen, um einen Agenten zu erstellen

Schritt	Aufgaben	Aktion
ANHANG Design Werkzeuge	Beschreiben	Fügen Sie eine klare Werkzeug-Workflow-Beschreibung (task, Eingänge, Ausgänge, Parameter)
	Parameter (fakultativ)	Konfigurationsknoten mit klaren Namen verwenden und Beschreibungen.
	Kommunikation Schicht (optional)	<a href="#">Hinzufügen</a> <a href="#">Tool-Nachrichtenausgang</a> um Text zurückzugeben Agent (liest erste Zeile, erste Spalte).
	Datenebene (fakultativ)	<a href="#">Hinzufügen</a> <a href="#">Workflow Input</a> / <a href="#">Ausgangsleistung</a> Knoten, wenn Daten benötigt werden durch das Werkzeug fließen.

<b>2. Werkzeug bauen</b> <b>Liste</b>	Workflow vorbereiten	Erstellen Sie einen separaten Agenten-Workflow, um Tools zu sammeln.
	Auf den Wunschzettel	Verwenden Sie die <a href="#">Dateien/Folders auflisten</a> Knoten, um den Ordner zu scannen mit Werkzeug-Workflows.
	Umrechnen	Verwendung <a href="#">Workflow zum Werkzeug</a> um Tool List zu generieren.
	Überprüfung	Überprüfen Sie Metadaten mit Icons: Beschreibung vorhanden, Parameter definiert, Datenports zugeordnet.
	Refreshing	Nach dem Ändern eines beliebigen Werkzeugs, speichern Sie Workflow und re-run <a href="#">Workflow zum Werkzeug</a> Knoten.
<b>3. Konfigurieren</b> <b>Agent</b>	<a href="#">Agent Prompter</a>	Systemnachricht, Benutzernachricht und Tool bereitstellen Liste.
	Datenports	Manuelles Hinzufügen von Eingabe-/Ausgabe-Ports, wenn Werkzeuge benötigen externe Daten.
<b>4. Optional</b> <b>Bereitstellung</b>	Interaktive Ansicht	Verwendung <a href="#">Agent Chat anzeigen</a> für Live-Konversationen.
	Bereitstellung	Wrap als KNIME Component und Einsatz über KNIME Business Hub.

## Beispiel: Aufbau eines Restaurantassistenten

Dieses Beispiel führt Sie durch den Prozess des Baus eines Restaurantassistenten mit die [KNIME AI Erweiterungen](#). Der Assistent soll das Restaurantpersonal durch die Handhabung unterstützen gängige Aufgaben durch einfache, sprachliche Sprache.

Jeder Schritt fügt ein neues Konzept hinzu, beginnend mit einem einfachen Werkzeug und schrittweise Einführung Parameter, bedingte Logik und Datenhandling.

Am Ende kann der Agent:

- Antworten Sie Fragen über Allergene mit

[Antworten auf Fragen über Allergene](#page42)

- Empfohlene alternative Buchungsoptionen mit

[Empfohlene alternative Buchungsoptionen](#page49)

## [Tool 1: Antwort Allergen Fragen \(Kommunikationsschicht\)](#)

Dieses erste Werkzeug führt die einfachste Art der Agens-Interaktion ein: ein Werkzeug, das nur die Kommunikationsschicht. Es erfordert keine Parameter und keine Dateneingabe.

Wenn ein Benutzer eine Frage zu Allergenen stellt, kann der Agent dieses Tool anrufen, um ein vordefinierter String mit Allergeninformationen. Der Agent nutzt diese Informationen dann formulieren ihre Antwort.

ANHANG Entwerfen Sie das Werkzeug

Der Werkzeug-Workflow enthält nur zwei Knoten:

- [Tabelle Schöpfer](#) Knotenpunkt

Verwenden Sie diesen Knoten, um eine einreihige, einkalige Tabelle mit Allergendetails für jede Menüpunkt.

Geben Sie den folgenden String als Tabelleninhalt ein:

```
Grillhähnchen: Gluten: Nein. Dairy: Nein. Nüsse: Nein, Shellfish: Nein, Fisch: Nein, Sesam: Nein
Salmon Teriyaki: Gluten: Nein. Dairy: Nein. Nüsse: Nein, Shellfish: Nein, Fisch: Ja, Sesam: Ja
Shrimp Tacos: Gluten: Nein. Dairy: Ja. Nüsse: Nein, Shellfish: Ja, Fisch: Nein, Sesam: Nein
Vegan Burger: Gluten: Ja. Dairy: Nein. Nüsse: Nein, Shellfish: Nein, Fisch: Nein, Sesam: Nein
Schokolade Kuchen: Gluten: Ja. Dairy: Ja. Nüsse: Ja, Shellfish: Nein, Fisch: Sesam: Nein
Pad Thai: Gluten: Nein. Dairy: Nein, Nüsse: Ja, Shellfish: Nein, Fisch: Sesam: Ja.
```

Der Agent wird diese Informationen benutzen, um Fragen wie:

„Führt das gegrillte Huhn

Sesam enthalten? „ oder „Welche Gerichte sind glutenfrei? „

- [Tool-Nachrichtenausgang](#) Knotenpunkt

Dieser Knoten wandelt den Tabelleninhalt in eine Nachricht um, die der Agent lesen kann.

Im Konfigurationsdialog den Parameternamen umbenennen Allergene . Das macht deutlicher für den Agenten zu verstehen, welche Art von Informationen zurückgegeben werden, insbesondere wenn mehrere Werkzeuge sind verfügbar.

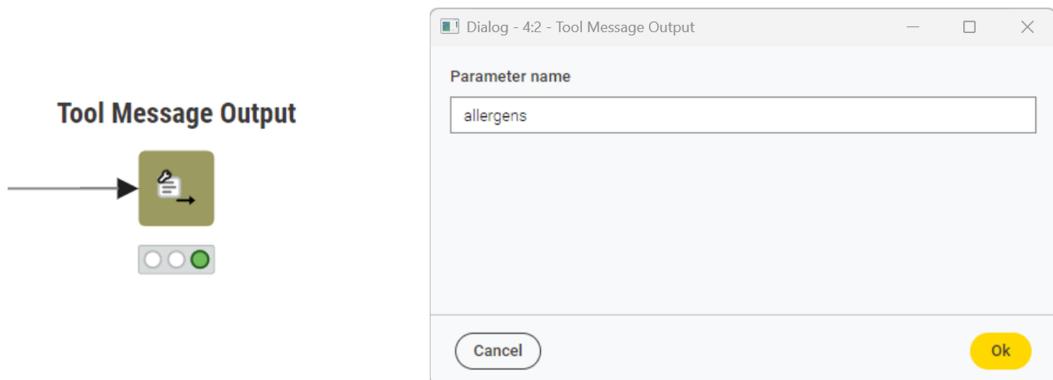


Abbildung 36: Die [Tool-Nachrichtenausgang](#) Konfiguration. Der Parameter wird auf Allergene umbenannt den Inhalt besser beschreiben.

## 2. Beschreiben Sie das Tool

Sobald die Workflow-Logik abgeschlossen ist, ist der letzte Schritt, das Werkzeug zu beschreiben, so dass der Agent weiß wenn es zu benutzen. Diese Beschreibung wird in der [Workflow Beschreibung](#) Feld, gefunden unter der [Info zum Thema Panel](#) in der KNIME Analytics Platform.

Der Agent wird sich auf diese Beschreibung verlassen, um zu entscheiden, ob das Tool mit der Frage eines Benutzers übereinstimmt. Je genauer und informativ der Text, desto wahrscheinlicher wird der Agent das Tool verwenden effektiv.

Verwenden Sie die folgenden:

```

Werkzeugname: allergens_information
Beschreibung: Dieses Tool gibt eine Reihe mit Informationen über das Restaurant zurück
Gerichte und ihre Allergene. Es kann Fragen wie:
"Enthält das gebrillte Huhn Sesam?"
```

Einmal hinzugefügt, wird diese Beschreibung Teil der Metadaten des Tools und wird abgeholt während der Registrierung über [Workflow zum Werkzeug](#) Knoten.

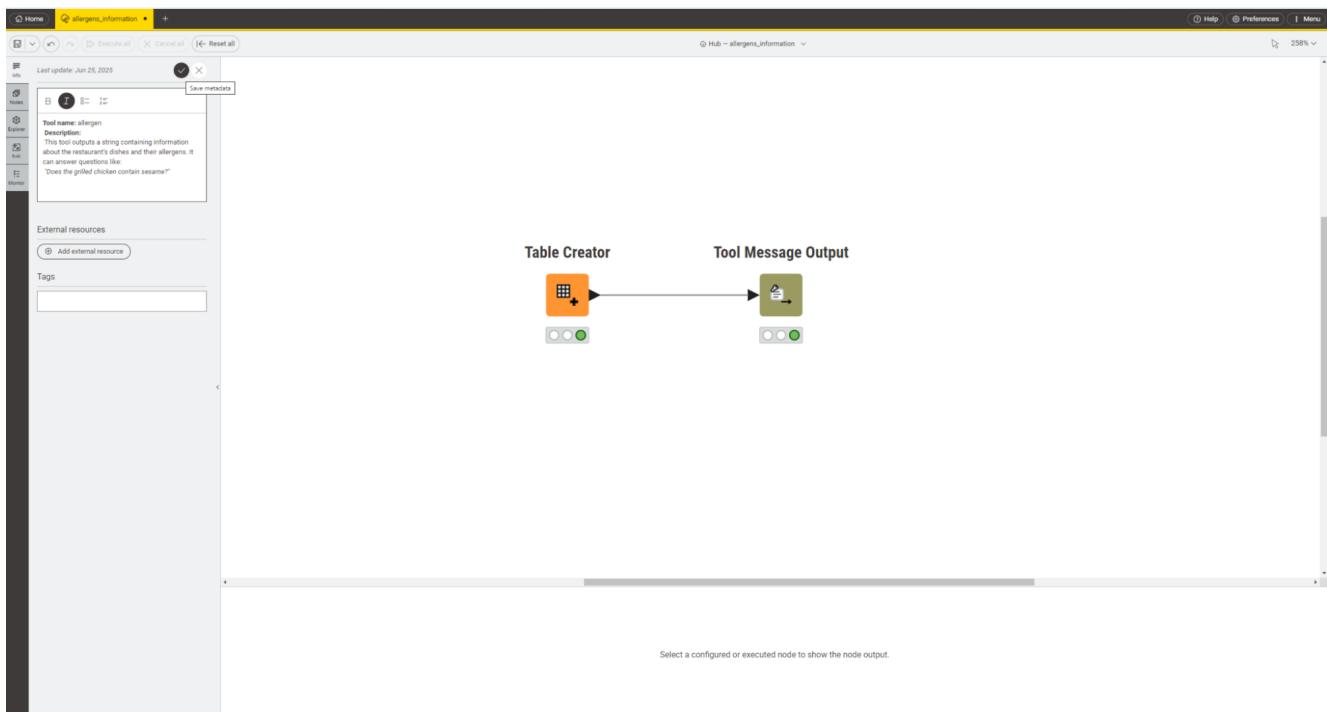


Abbildung 37. [Werkzeug-Workflow](#) jetzt hat eine Beschreibung

### [Tool 2: Handle Booking Requests \(Parameter\)](#)

Dieses Tool führt parametrierte Toolanrufe ein. Der Agent liest die Benutzeranforderung (z. Anzahl der Personen und das gewünschte Datum), extrahiert diese Informationen und sendet sie als Parameter zum Werkzeug-Workflow.

ANHANG Parameter zum Tool hinzufügen

Dieses Tool verwendet zwei Parameter:

- Anzahl : die Anzahl der Sitze, die der Benutzer buchen möchte
- Zurück zur Übersicht : das gewünschte Datum für die Reservierung

Indem Sie jedem Parameter einen klaren Namen und eine kurze Beschreibung geben, helfen Sie dem Agenten verstehen, welche Werte aus der Anfrage des Nutzers zu extrahieren sind.

#### **Beispiel:**

Benutzereingabe: Ich brauche einen Tisch für zwei Personen für 6/25/2025.

Ausgewählte Parameter:

- Anzahl : 2

- Zurück zur Übersicht : 2025-06-25

## 2. Überprüfen Sie den Datensatz

Das Tool arbeitet mit einer Tabelle der aktuellen Verfügbarkeit, gespeichert in einer Datei namens `restauant_reservations.csv`.

Die Tabelle enthält:

Tabelle ID	Sitze	Datum	Zeit	Verfügbar
T1	2.	2025-06-24	19:	Ja.
T2	ANH ANG	2025-06-24	19:	Ja.
T3	6	2025-06-24	19:	Nein
T4	ANH ANG	2025-06-24	20 Uhr	Ja.
T5	2.	2025-06-24	21 Uhr	Ja.

## 3. Entwerfen Sie das Werkzeug

### Parameter konfigurieren

Verwendung :

- [Integer Konfiguration](#) zu sammeln Anzahl
- [Datum und Uhrzeit Konfiguration](#) zu sammeln Zurück zur Übersicht

Stellen Sie sicher, dass beide Parameterknoten enthalten **deskriptive Etiketten** und **kurze Erläuterungen**. Das hilft dem Agenten zu verstehen, welche Informationen zu übergeben.

Zusammenführen der beiden Parameter mit [VerschmelzungsvARIABLES](#) sie in der Filterlogik verwenden.

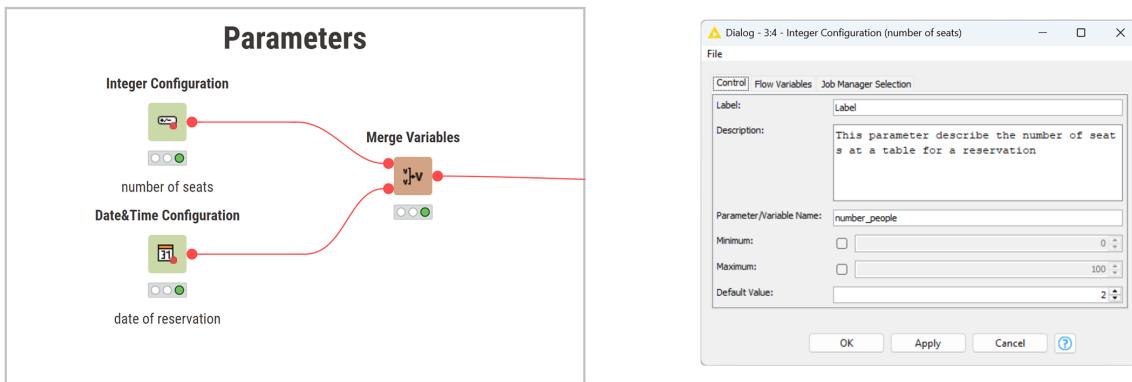


Abbildung 38. Der Konfigurationsdialog der

[Integer Konfiguration](#)

Knotenpunkt

**Filterergebnisse**Verwenden eines [Leerer Tischschalter](#)

Knoten für zwei Pfade:

- Wenn passende Tabellen gefunden werden:

- [Top K Row Filter](#) node wählt eine verfügbare Tabelle aus.
- [Update .csv Metanode](#) aktualisiert die Datei restaurant\_reservations.csv durch Überschreiben es, die Verfügbarkeit der Tabelle von "Ja" zu "Nein" ändern, um die Buchung zu registrieren.
- [Ausdruck](#) eine Bestätigungs Nachricht erstellt:

```
string("Die Buchung für Tabelle " + ${"Table ID"} +
" mit " + ${"Seats"} + " Menschen, auf " + ${"Date"} +
" wurde bestätigt!")
```

- [Spaltenfilter](#) hält nur die Nachrichtenspalte

- Wenn keine Tabelle verfügbar ist:

- [Tabelle Schöpfer](#) erstellt die Fallback-Nachricht: "Keine Tabellen stehen für gewünschtes Datum."
- [Spaltenfilter](#) hält nur die Nachrichtenspalte.

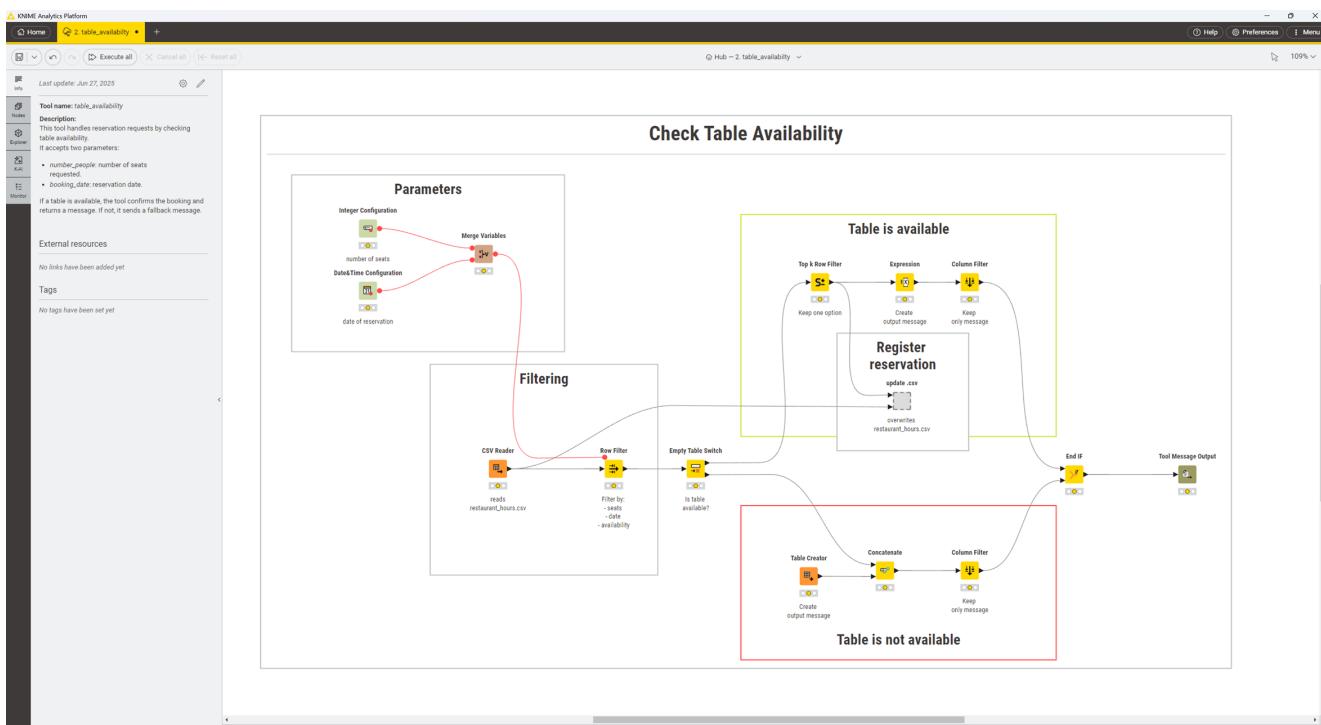
Verbinden Sie die beiden Zweige mit einem [END IF](#) um ein einziges Ergebnis zurückzugeben.

**Konfigurieren der Kommunikationsebene**

Verwenden Sie die [Tool-Nachrichtenausgang](#) die letzte Nachricht an den Agenten senden. Der Parameter Name ist eingestellt Tabelle\_Verfügbarkeit den Inhalt klar beschreiben.

Das Tool gibt nun entweder eine Buchungsbestätigung oder eine Nichtverfügbarkeitsnachricht, basierend auf

die Eingabe.



**Abbildung 39. Die Tabelle Verfügbarkeit** Werkzeug-Workflow mit zwei Pfaden: verfügbar oder nicht verfügbar, enden in einem einzigen Nachrichtenausgang.

L 347 vom 20.12.2013, S. 1). Beschreiben Sie das Tool

Sobald der Werkzeug-Workflow abgeschlossen ist, fügen Sie eine Beschreibung in der der Agent verstehen, wann es zu verwenden.

Infotafel zum Thema Workflow Dies hilft

Verwenden Sie die folgenden:

Werkzeugname: table\_Verfügbarkeit

Beschreibung: Dieses Tool behandelt Reservierungsanfragen durch Überprüfung der Tischverfügbarkeit.

Es akzeptiert zwei Parameter:

- Nummer\_Personen: Anzahl der Plätze.
- booking\_date: Reservierungsdatum.

Wenn eine Tabelle verfügbar ist, bestätigt das Tool die Buchung und gibt eine Nachricht zurück. Wenn nicht, es sendet eine Rückmeldung.

### [Tool 3: Empfohlene alternative Buchungstermine \(Concatenate tools\)](#)

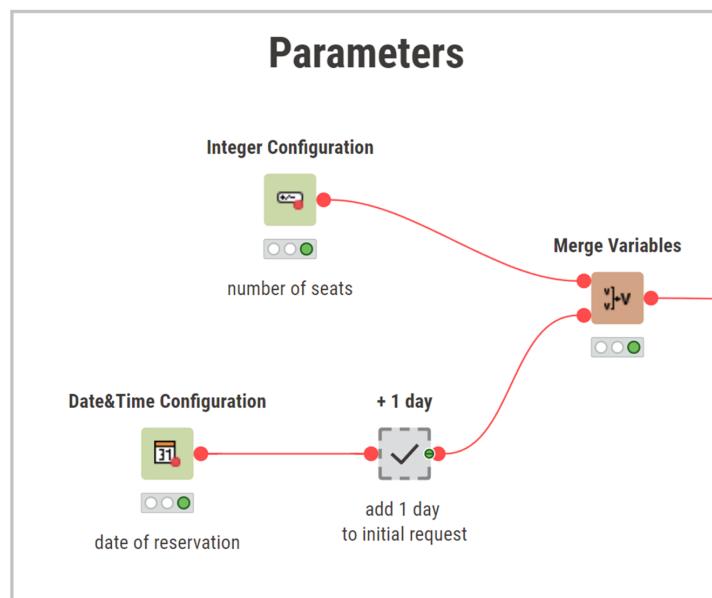
Dieses Tool baut auf dem vorherigen, indem es einen Fallback bietet. Wenn keine Tabellen verfügbar sind angefordertes Datum, kann der Agent dieses Tool verwenden, um die Verfügbarkeit für den nächsten Tag zu überprüfen und vorzuschlagen eine Alternative.

ANHANG Parameter hinzufügen

Das Tool verwendet die gleichen Parameter wie Tool 2:

- Anzahl
- Zurück zur Übersicht

Verwenden Sie Konfigurationsknoten, um diese Werte zu erfassen. Dann wenden Sie eine [Datum und Uhrzeit](#) um die Zurück zur Übersicht einen Tag nach vorn.



[Abbildung 40. Die Parameterkonfiguration](#)

[Ausgewählte Alternative](#)

[Werkzeug. ADatum und Uhrzeit](#)

in den  
Warenkorb

eines Tages zum gewünschten Buchungsdatum, um am folgenden Tag nach Verfügbarkeit zu suchen.

## 2. Workflow Design

Der Workflow ist ähnlich zu Tool 2, mit einem entscheidenden Unterschied: er prüft die Tischverfügbarkeit für den Tag nach der ursprünglichen Anfrage. Wenn eine alternative Tabelle gefunden wird, gibt das Tool einen Vorschlag wie: „Es gibt eine Alternative für den Tag, nach dem Tisch T2 mit 4 Personen, auf 2025-06-25 ist frei. „, nichts ist verfügbar, das Werkzeug gibt eine Rückfall-Nachricht zurück. Dieses Tool bestätigt nicht

wenn

Buchungen. Es schlägt nur Alternativen vor.

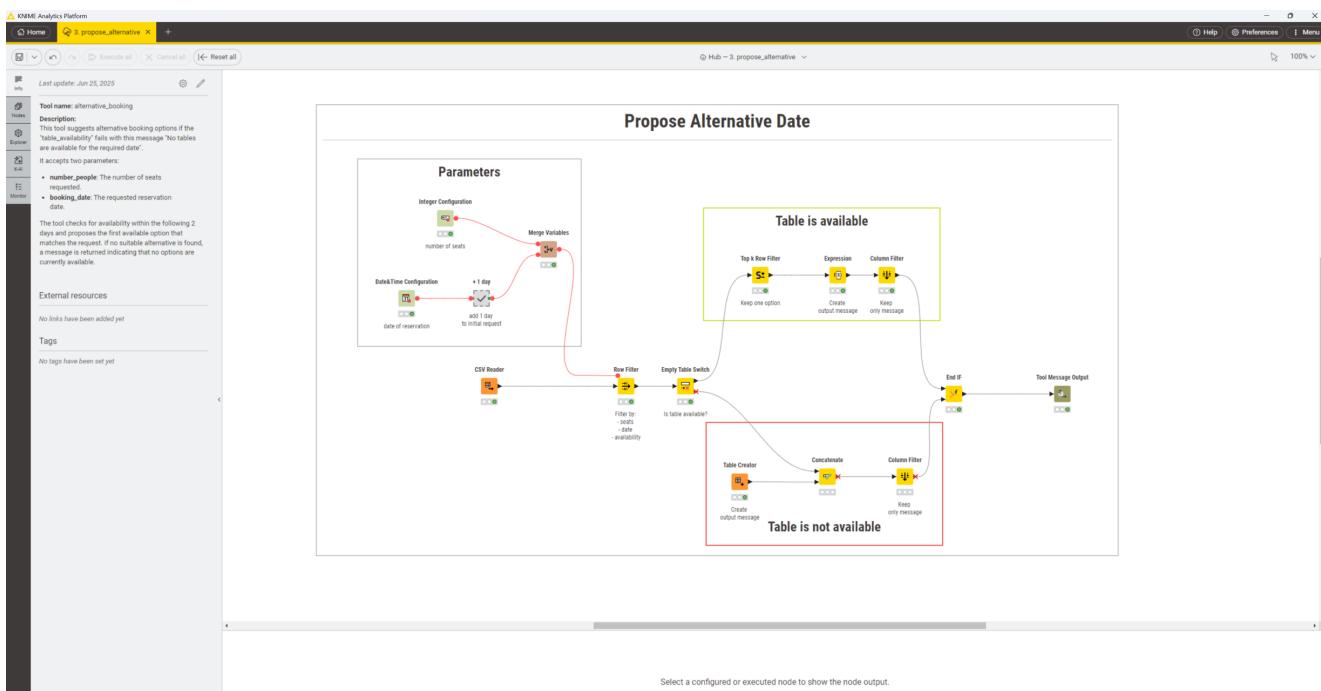


Abbildung 41. [Die richtige Alternative](#)

Tool Workflow, der ein alternatives Reservierungsdatum vorschlägt

zum Benutzer

### 3. Kommunikationsschicht

Verwenden Sie die [Tool-Nachrichtenausgang](#) um die Nachricht zurückzugeben. Setzen Sie den Parameternamen ein `alternative_Buchung`.

L 347 vom 20.12.2013, S. 1). Beschreiben Sie das Tool

Gehen Sie zum Info zum Thema [Panel und hinzufügen:](#)

Werkzeugname: `alternative_booking`

Beschreibung: Dieses Tool schlägt alternative Buchungsoptionen vor, wenn das gewünschte Datum ist ausgebucht. Es prüft die Verfügbarkeit am folgenden Tag und gibt einen Vorschlag zurück wenn ein offener Tisch gefunden wird.

### Tool 4: Analyze Customer Review Sentiment (Data Layer)

Dieses Tool stellt die **Datenschicht**. Es verarbeitet eine Tabelle von Kundenbewertungen, analysiert die Stimmung jeder Überprüfung mit einem LLM und gibt zurück:

- Eine kurze Nachricht mit der Anzahl der positiven und negativen Bewertungen.
- Eine Datentabelle, in der jede Überprüfung als positiv oder negativ bezeichnet wird.

#### ANHANG Dateneingabe definieren

Dieses Tool erhält eine Tabelle mit einer Spalte mit dem Namen Review, die benutzerdefiniertes Feedback enthält wie:

Überprüfung
Das Essen war erstaunlich, großartiger Service!
Schreckliche Erfahrung. Langes Warten und kaltes Essen.

Damit der Agent diese Tabelle in das Werkzeug übergibt, benötigen Sie eine definiert, was das Werkzeug in der Datenschicht erwartet. Der Agent selbst kann die Daten: es löst nur das Werkzeug aus und liest die resultierende Nachricht.

#### [Workflow Input](#)

Knoten. Dieser Knoten

Es ist nützlich, einen kleinen mock-Datensatz (z.B. mit einem so können Sie die Logik des Tools testen. Diese Mock-Daten werden nur verwendet, wenn das Werkzeug alleine ausgeführt wird. Beim Aufruf durch den Agenten wird der Eingang durch die in der Agentischer Workflow.

#### [Tabelle Schöpfer](#) ) während der Entwicklung

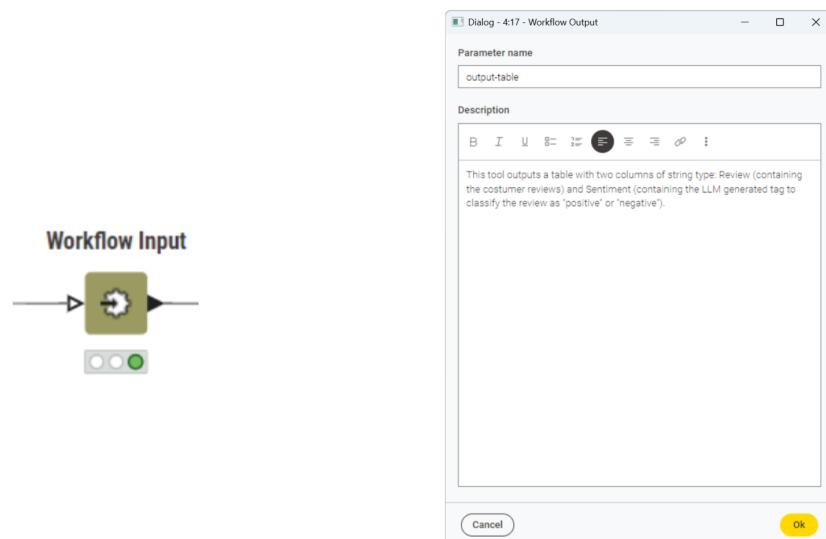


Abbildung 42: Die [Workflow Input](#) Konfigurationsdialog

## 2. Workflow Design

Der Werkzeugarbeitsablauf weist eine Datenschicht und eine Kommunikationsschicht auf.

### Datenebene

Die Datenschicht behandelt die Aufnahme und Transformation externer Daten:

- [Workflow Input](#)

Erhält die Eingabetabelle vom Agenten. Der Eingang muss eine einzelne Spalte mit dem Namen enthalten Überprüfung.

- [Ausdruck](#)

Erstellt eine Aufforderung für jede Bewertung mit:

```
„string("Ist diese Bewertung positiv oder negativ? nur ein Etikett in Kleinbuchstaben zurückgeben:  
" + $["Review"]"
```

- [LLM Promter](#)

Senden Sie die Aufforderung an ein ausgewähltes Modell (z.B. GPT-4.1-nano) um die Stimmung zu klassifizieren und gibt die Vorhersagen des Modells als eine neue Spalte namens Sentiment aus.

- [Workflow-Ausgang](#)

Die Prognosen des Modells werden der Eingabetabelle als neue Sentiment-Spalte beigefügt.

Dieser angereicherte Datensatz wird bei Bedarf an den Agenten zurückgeschickt.

### Kommunikationsebene

die Kommunikationsschicht baut eine natürliche Sprachnachricht, die der Agent mit:

- [Wertzähler](#)

Zählt, wie viele Bewertungen in jede Stimmungskategorie fallen (z.B. positiv, negativ).

- [Tabelle Transposer](#)

Konvertiert Zählungen in ein Zeilenformat, so dass eine einzelne Nachricht daraus gebaut werden kann.

- [Ausdruck](#)

erzeugt einen Nachrichtenstring wie:

„Es gibt 15 positive Überprüfungen und 5 negative Überprüfungen.“

- [Spaltenfilter](#)

hält nur die Nachrichtenspalte (erste Zeile, erste Zelle erforderlich von

[Tool-Nachrichtenausgang](#)

)

- [Tool-Nachrichtenausgang](#)

Sendet die letzte Zusammenfassungsmeldung an den Agenten. Der Parameter ist benannt review\_summary.

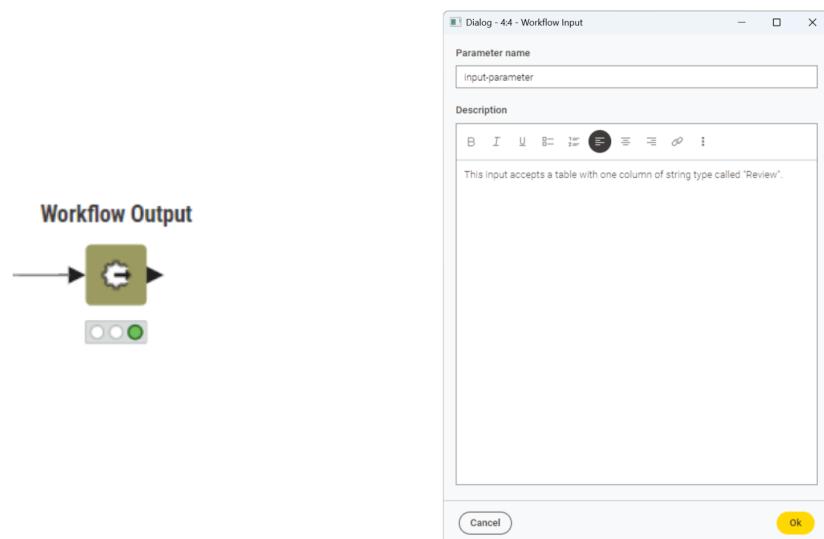
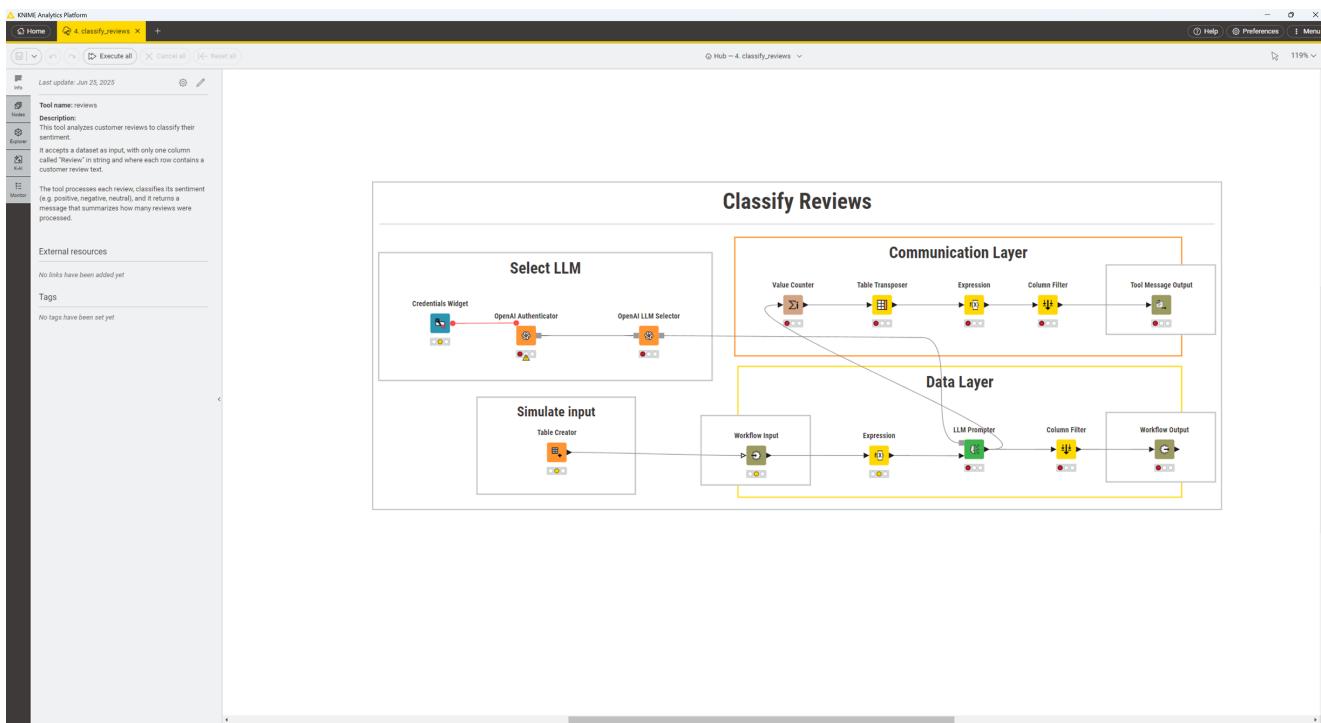


Abbildung 43. [Workflow-Ausgang](#) Knotenkonfiguration



**Abbildung 44.** Die [Bewertungen einordnen](#) Tool-Workflow, der Bewertungen mit einem LLM klassifiziert.

Der Agent liest nur die Nachricht von [Tool-Nachrichtenausgang](#). Wenn nötig, die bereicherte Tabelle ist als Datenausgabe für andere Werkzeuge verfügbar.

### 3. Beschreiben Sie das Tool

Öffne die [Info zum Thema](#) Panel und hinzufügen:

Werkzeugname: classify\_reviews

Beschreibung: Dieses Tool analysiert Kundenbewertungen, um ihre Einschätzung zu klassifizieren. Es akzeptiert einen Datensatz mit einer Spalte namens Review, mit Text. Jede Bewertung wird als entweder positiv oder negativ.

Das Werkzeug kehrt zurück:

- Eine Zusammenfassungsnachricht, in der angezeigt wird, wie viele Bewertungen als positiv oder negativ eingestuft wurden.
- Eine transformierte Tabelle mit einer neuen Sentiment-Säule.

### [Letzte Schritte: Verbinden und führen Sie Ihren Agenten](#)

Mit allen vier Werkzeugen komplett, Ihr Agent ist bereit, zu verdanken, Trigger-Tools und Rückkehr hilfreich

Antworten basierend auf Benutzeranfragen.

ANHANG Alle Werkzeuge eintragen

Stellen Sie diese Tool-Workflows in einem einzigen Ordner namens

Werkzeuge:

- Werkzeuge/allergene\_Informationen
- Werkzeuge/Tabelle\_Verfügbarkeit
- Werkzeuge/alternative\_Buchung
- Tools/classify\_reviews

2. Tool List Workflow erstellen

In einem neuen Workflow erstellen Sie die Werkzeugliste:

- [Dateien/Folders auflisten](#)
  - Konfigurieren Sie es, um auf den Werkzeugordner zu zeigen
  - Dies ruft alle .knwf Werkzeugarbeitsabläufe

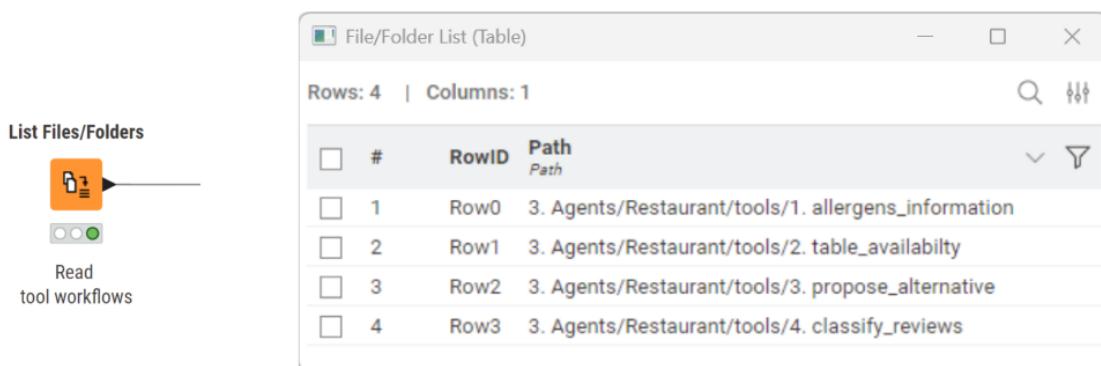


Abbildung 45. Die [Dateien/Folders auflisten](#) node liest alle Tool-Workflows aus den ausgewählten "tools" Verzeichnis.

- [Workflow zum Werkzeug](#)

- Dies wandelt jeden Workflow in ein Werkzeugobjekt mit zugehörigen Metadaten um
- Icons geben an, ob das Tool Parameter, Datenports enthält oder eine Beschreibung

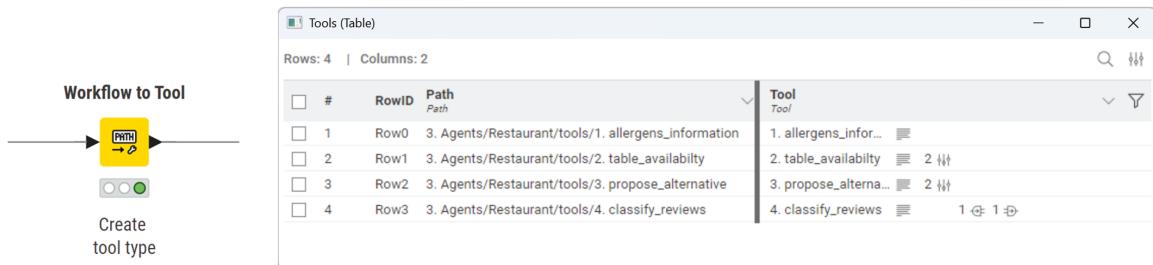


Abbildung 46. Der Ausgang der [Workflow zum Werkzeug](#) node zeigt Icons, die helfen, das Tool zu überprüfen Beschreibungen, Parameter und Dateneingänge/Ausgänge.

### 3. Aufbau des Agenten-Workflows

- Fügen Sie die [Agent Prompter](#) node und set this as Systemnachricht :

Sie sind ein Restaurantassistent.  
 Denken Sie immer weiter, bis die Anfrage des Benutzers vollständig bearbeitet wird.  
 Verwenden Sie Ihre verfügbaren Tools, um Daten zu überprüfen und Entscheidungen zu treffen. Nicht erraten.  
 Wenn eine Reservierungsanfrage eingegangen ist:  
 - Probieren Sie eine Tabelle direkt.  
 - Wenn nicht verfügbar, suchen Sie nach alternativen Terminen.  
 - Ja. Wenn keine Alternativen vorhanden sind, antworten Sie entsprechend.  
 Wenn Allergenfragen gestellt werden, verwenden Sie das Allergen-Tool, um die notwendigen Informationen.  
 Wenn Kundenbewertungen zur Verfügung gestellt werden, analysieren Sie ihre Einschätzung und berichten, wie viele wurden bearbeitet.

- Optional die Vorfüllung Benutzernachricht Feld (z. Können Sie einen Tisch für zwei Personen buchen  
 26. Juni? )
- In der Werkzeugspalte die Ausgabe der [Workflow zum Werkzeug](#) Knotenpunkt

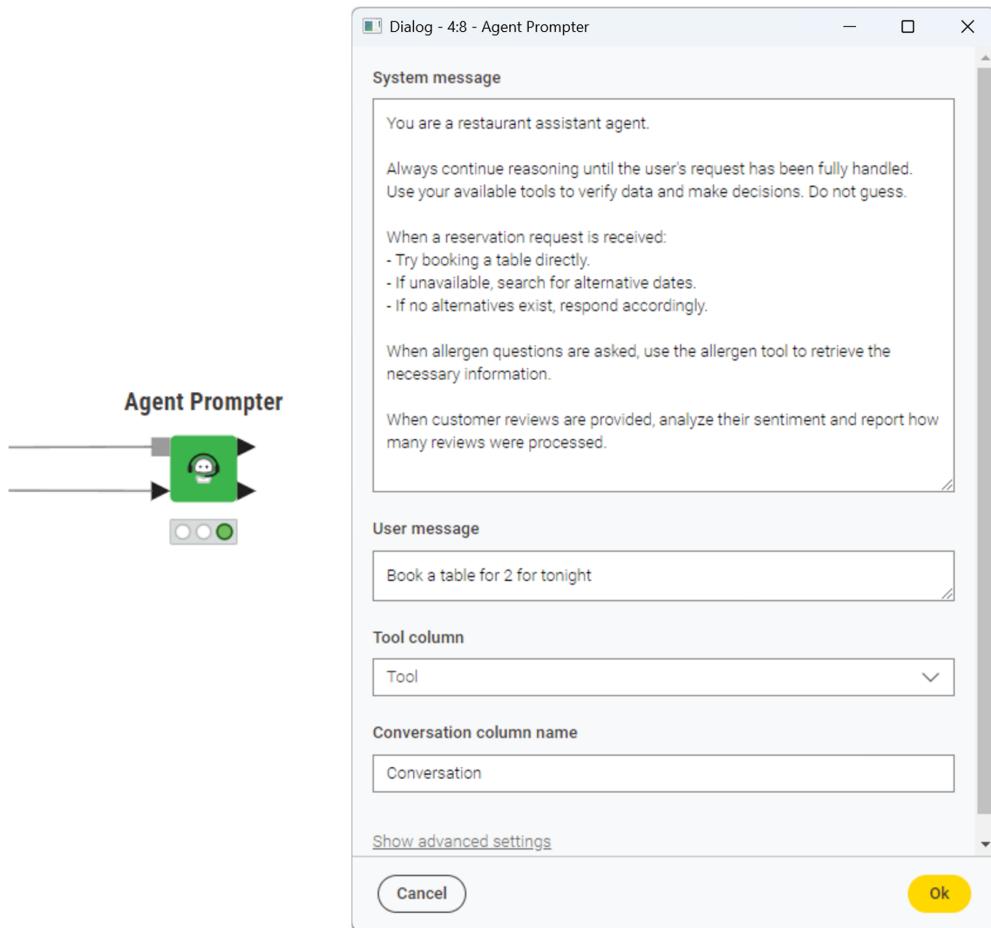


Abbildung 47: Die [Agent Prompter](#) Konfigurationsdialog

- Datenports aktivieren (für Werkzeug 4)

Damit der Agent mit externen Daten arbeiten kann (z.B. Kundenrezensionen für Stimmung

Analyse), Sie müssen Dateneingabe und Ausgabeports zu den [Agent Prompter](#) Node:

ANHANG Importieren Sie Ihren Datensatz mit einem[CSV Reader](#) Knoten.

Dies sollte eine Spalte enthalten, die Überprüfung, mit einer Rezension pro Zeile.

2. Rechtsklicken Sie auf die[Agent Prompter](#) Knoten.

Wählen Input Port hinzufügen und Ausgabeport hinzufügen aus dem Kontextmenü. Dies ermöglicht der Agent, Daten über die Werkzeuge zu empfangen und zu verarbeiten.

3. Verbinden Sie die [CSV Reader](#) zum Eingangsport des [Agent Prompter](#).

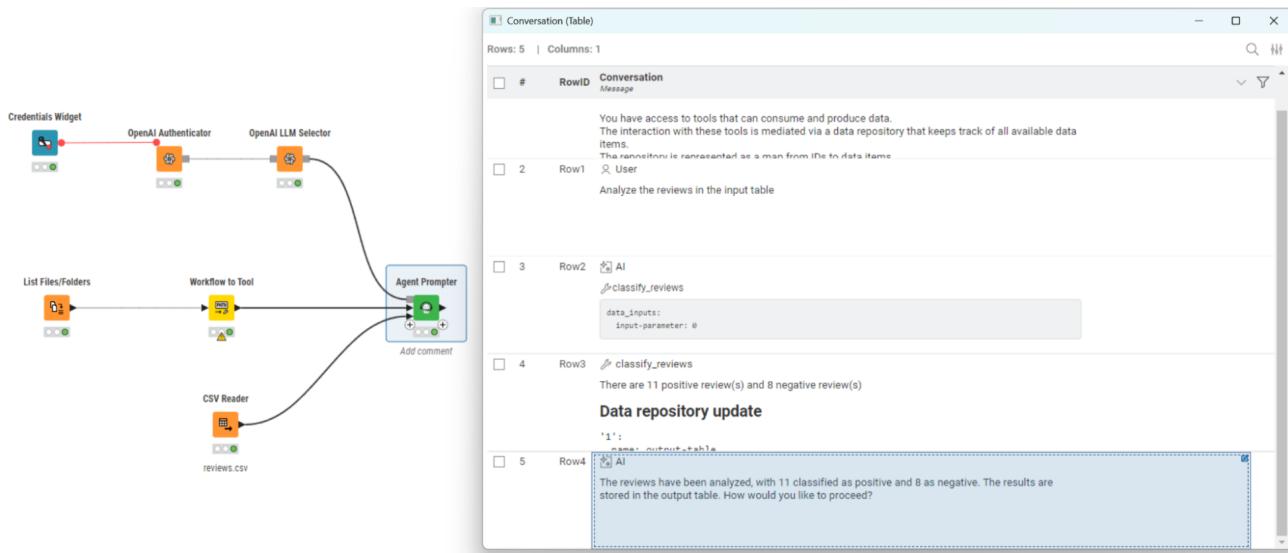


Abbildung 48. [Agent Prompter](#) mit einem zusätzlichen Ausgangsport: der Agent kann nun Daten zusammen mit seine Antwort.

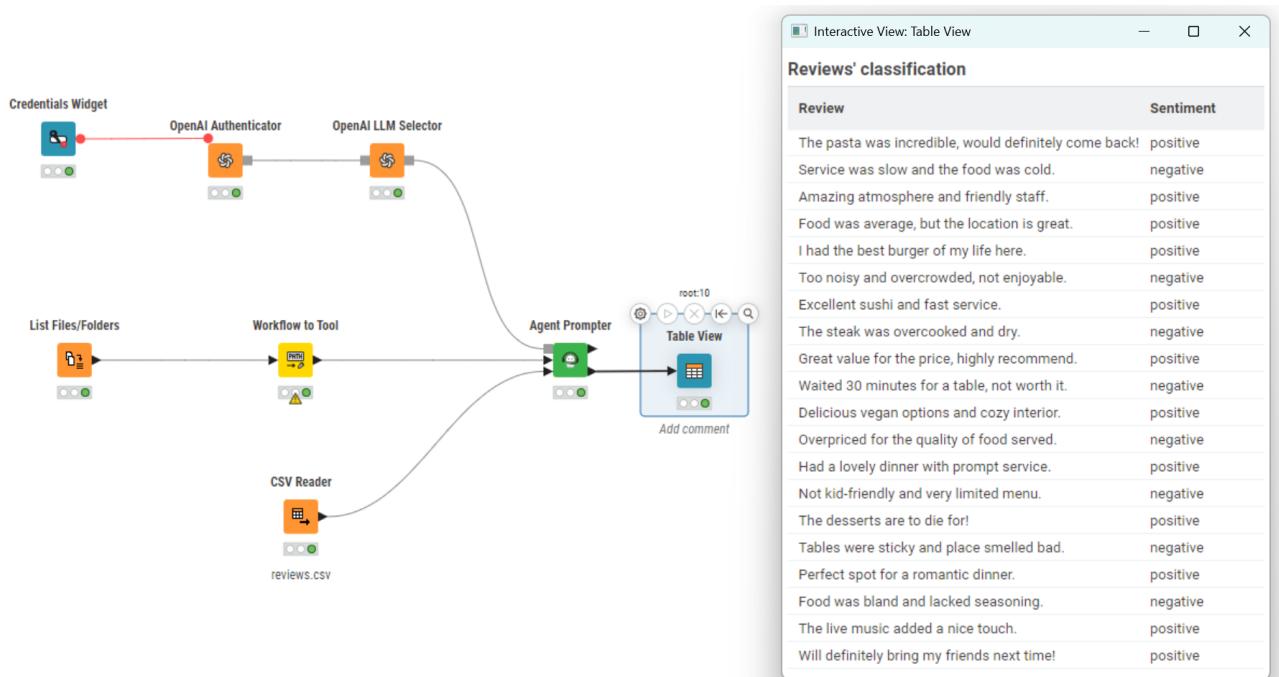


Abbildung 49. Die [Agent Prompter](#) hat einen Eingang und einen Ausgangsport: nur die Kommunikation Eine Schicht ist für den Benutzer sichtbar, keine Datenausgabe wird zurückgegeben.

#### 4. Laufen und Inspect

Lauf den Workflow. Dies ist ein interaktiver Prozess und kann nicht perfekt auf den ersten Versuch arbeiten.

Um Fehler zu beheben, verwenden Sie die **Debug-Modus** in der [Agent Prompter](#) Blick. Dieser Modus hilft Ihnen zu sehen die Argumentation des Agenten Schritt für Schritt.

Nach Abschluss dieses Prozesses, die [Agent Prompter](#) gibt ein Gespräch zwischen Benutzer, KI und Tools.

#	RowID	Conversation
		Message
1	Row0	User <b>Data Tools Interface</b> You have access to tools that can consume and produce data. The interaction with these tools is mediated via a data repository that keeps track of all available data items. The Data Tools Interface is represented as a man from the tv data item.
2	Row1	User Book a table for two people for the 2025.6.6
3	Row2	AI <b>table_availability</b> configuration: booking_date=2; "2025-06-26T00:00:00+02:00[Europe/Berlin]" number_people=4; 2
4	Row3	AI <b>table_availability</b> The booking for table T1 with 2 people, on 2025-06-26 was confirmed!
5	Row4	AI The table for two people has been successfully booked for June 26, 2025.

Abbildung 50. Die Ausgangsansicht der [Agent Prompter](#) Keine Nachricht

Wenn ein Tool ausfällt, zum Beispiel weil ein Sprachmodell keine Anmeldeinformationen enthält, wird die Debug-Track wird deutlich zeigen, wo der Fehler passiert ist. Dies erleichtert die Identifizierung und Fixierung der Problem.

## 5. Chat-Interface hinzufügen

Um den Assistenten für Endbenutzer interaktiv zu machen, verwenden Sie die [Agent Chat anzeigen](#) Knoten.

Dazu:

ANHANG Fügen Sie die [Agent Chat anzeigen](#) Knoten zu Ihrem Workflow.

2. Schließen Sie die Ausgabe der Werkzeugliste von der [Workflow zum Werkzeug](#) Knoten.

3. Wenn Ihr Agent externe Daten verwendet, verbinden Sie auch die entsprechenden Eingabetabellen.

Sobald Sie konfiguriert sind, können Sie den Workflow in eine Komponente einpacken und über

[KNIME](#)

[Business Hub](#).

Damit ist Ihr Assistent als Dienst erreichbar, bereit, Benutzeranfragen zu erhalten und zurückzukehren werkzeugbasierte Antworten.

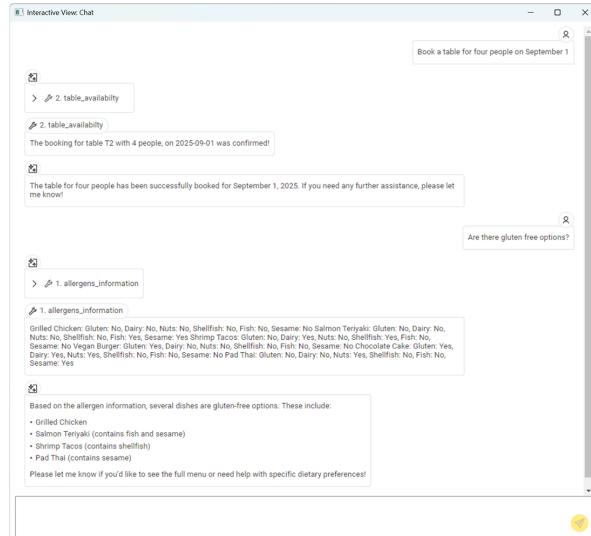


Abbildung 51. Die [Agent Chat anzeigen](#) bietet eine Live-Konversationsschnittstelle.

# AI Governance

## GPT4All (Lokale Modelle)

KNIME unterstützt die lokale Ausführung von Open-Source-Modellen durch **GPT4All**, so dass Sie laufen große Sprachmodelle (LLMs) und Einbettmodelle direkt auf Ihrer Maschine. Das ermöglicht volle Offline-Betrieb, entfernt Abhängigkeit von externen APIs, schützt Privatsphäre-sensible Daten und eliminiert nutzungsbasierte Kosten im Zusammenhang mit kostenpflichtigen Anbietern.

**Systemvoraussetzung:** GPT4All-Knoten können nicht mit Red Hat 8 kompatibel sein.  
Bitte überprüfen Sie Ihr Betriebssystem, bevor Sie diese Knoten verwenden.

### Hauptmerkmale

- **Keine externen APIs**

GPT4 Alle laufen voll auf Ihrer lokalen Hardware. Keine Internetverbindung oder externe Dienste (wie OpenAI oder Hugging Face) sind erforderlich.

- **Keine Authentifizierung erforderlich**

Da Modelle lokal ausgeführt werden, sind keine Authenticator-Knoten oder API-Keys erforderlich.

- **Open-Source-Modelle**

Sie können aus einer Vielzahl von gemeindeerhaltenen Modellen wählen.

### Modellaufbau

Bevor Sie GPT4All Modelle in KNIME verwenden, müssen Sie die Modelldateien erhalten:

ANHANG Modelle von Hugging Face Hub herunterladen, die im Format .gguf verfügbar sind.

(z. NousResearch/Nous-Hermes-llama2-GGUF )

2. Speichern Sie die Modelldatei lokal auf Ihrer Maschine.

3. Geben Sie in der Connector-Knotenkonfiguration den vollen Dateipfad zum heruntergeladenen .gguf an Modelldatei.

## Hardware-Konfiguration

Sie können wählen, mit welcher Verarbeitungseinheit das Modell ausgeführt werden soll:

- Cpu verwendet den zentralen Prozessor des Systems (Standardeinstellung).
- Gpu verwendet die besten verfügbaren GPU, unabhängig von Anbieter.
- Amid , Nvidia , oder intel wählen Sie einen bestimmten Anbieter.
- Spezifischer GPU-Name läuft auf einer bestimmten GPU, wenn mehrere verfügbar sind und richtig konfiguriert.

Die Auswahl einer GPU kann die Inferenzgeschwindigkeit für größere Modelle deutlich verbessern.

## GPT4All Connector Nodes

Die [KNIME AI Erweiterung](#) beinhaltet dedizierte Anschlussknoten für GPT4All Modelle:

- [Lokale GPT4All LLM Connector](#)
- [GPT4All Embedding Connector](#)

## Beispiel-Workflow

Für Unternehmensnutzer, die an zentraler Modelführung und Zugang interessiert sind  
Steuerung, KNIME Business Hub unterstützt auch GenAI Gateway. Diese Funktion ermöglicht  
Admins zur Konfiguration und Verwaltung von Chat- und Einbettungsmodellen zentral über  
eine Organisation. Weitere Einzelheiten finden Sie in der [Business Hub Admin Guide](#).



KNIME AG  
Talacker 50  
8001 Zürich, Schweiz  
[www.knime.com](http://www.knime.com)  
[Info@knime.com](mailto:Info@knime.com)