



Jev entscheidet. Es schreibt nicht.

Jev von TypeSafe AI ist kein Chatbot. Du gibst ihm einen Zustand – zum Beispiel eine Kundenanfrage – und vorher festgelegte Fragen. Zurück kommen Entscheidungen und Wahrscheinlichkeiten, kein formulierter Text.

Ideal für wiederkehrende Entscheidungen in Software.
Nicht gedacht zum Schreiben, Erklären oder für offene Gespräche.

Choice

Wählt eine Option aus deiner Liste, etwa Support, Verkauf oder Buchhaltung.

Score

Ordnet etwas auf einer beschriebenen Skala ein, etwa niedrig bis kritisch.

Noul

Liefert die Wahrscheinlichkeit für eine Ja-/Nein-Aussage.

Herstellerangaben: bis zu 193,6× schneller und 444,6× günstiger in vier eigenen Workflows. Das sind keine allgemeinen Garantien. Aktueller Listenpreis: 0,042 US-Dollar je 1 Mio. Eingabetokens; Ausgabetokens kostenlos.

[TypeSafe öffnen](#) • Quellen: S1 • S2 • S4



Der Unterschied in einem Blick.

ChatGPT / Claude Erzeugen Text Token für Token. Gut für Erklärungen, Entwürfe, Recherchegespräche und offene Aufgaben.

Jev Bewertet alle vorgegebenen Fragen parallel. Gut für Klassifizieren, Priorisieren, Prüfen und Weiterleiten.

PASST GUT

Du möchtest täglich E-Mails zuordnen, Leads priorisieren oder einen Workflow wählen. Text oder JSON kommt hinein; die erlaubten Antworten stehen vorher fest.

PASST NICHT

Du brauchst einen Social-Post, eine Zusammenfassung, Rechnungen, Bilder oder offene Beratung. Dafür bleibt ein Chat- oder Reasoning-Modell zuständig.

Jev wählt aus deinem Menü. Es erfindet nicht das Menü. Typsicherheit verhindert ungültige Formen – nicht falsche Entscheidungen.



01 / Ohne Code im Playground testen.

01 Playground öffnen

Melde dich in der TypeSafe-Konsole an. Der Direktzugang befindet sich derzeit im Early Access.

02 State einfügen

Nutze eine echte, anonymisierte Nachricht oder diesen Mustertext: „Die Abbuchung ist doppelt. Bitte heute prüfen, ich brauche das Geld zurück.“

03 Fragen anlegen

Choice: Abteilung mit billing, support, sales. Noul: „Ist die Nachricht zeitkritisch?“ Score: ruhig, frustriert, sehr verärgert.

04 Ergebnis lesen

Prüfe Auswahl, Wahrscheinlichkeiten und Confidence. Niedrige Sicherheit ist ein Signal für menschliche Prüfung – nicht zum Raten.

- ☐ Die Antwortoptionen schließen sich verständlich voneinander ab.
- ☐ Eine Option „sonstiges“ fängt unbekannte Fälle ab.
- ☐ Fünf typische und fünf schwierige Beispiele wurden geprüft.

Englisch ist laut TypeSafe die primäre Trainingssprache. Teste deutsche Inhalte deshalb mit eigenen Beispielen, bevor du sie automatisierst.



02 / Eine echte Entscheidung bauen.

Installiere zuerst den offiziellen TypeSafe-Skill in deinem Coding-Agenten. Lege deinen API-Key als TYPESAFE_API_KEY in der vorgesehenen Umgebungsverwaltung ab – niemals direkt im Prompt oder Code.

Rolle: Du bist mein Entwickler für eine kleine, überprüfbare Jev-Automation.

Kontext: Ich erhalte wiederkehrende [ART DER INHALTE]. Vorhanden sind [DATEI/ORDNER/QUELLE] und die erlaubten Entscheidungen [OPTIONEN]. Der TypeSafe-Skill ist installiert; TYPESAFE_API_KEY ist sicher gesetzt.

Aufgabe: Baue ein minimales lokales Programm, das jeden Eintrag mit Jev 1.13 bewertet und genau eine erlaubte Option plus Wahrscheinlichkeiten ausgibt. Lege Fragen, Kriterien und Schwellenwerte zentral und lesbar ab. Bei niedriger Sicherheit oder „sonstiges“ muss das Programm an mich übergeben statt automatisch zu handeln.

Ergebnis: Quellcode, kurze Startanleitung, Beispieldatei mit klar als fiktiv markierten Daten und Ergebnisprotokoll. Keine irreversible Aktion, kein Versand und keine Änderung der Quelle.

Prüfung: Erstelle mindestens 10 typische und 10 schwierige Testfälle mit erwarteten Labels. Führe die Tests tatsächlich aus, vergleiche Erwartung und Ergebnis, verbessere Fragen oder Grenzen und teste erneut. Prüfe abschließend aus Sicht eines unbeteiligten Nutzers: Sind Optionen, Unsicherheit, Kosten und nächster Schritt verständlich? Trenne echte Testergebnisse von offenen Punkten.

Bevor Jev automatisch handelt.

VIER SICHERHEITSREGELN

Klein zerlegen	Eine Frage pro Urteil. Breite Bewertungen in getrennte Faktoren aufteilen.
Unsicherheit nutzen	Schwellenwerte mit echten Beispielen festlegen. Darunter entscheidet ein Mensch.
Version festhalten	Für stabile Abläufe jev-1.13.0 pinnen; jev-latest kann sich später ändern.
Folgen begrenzen	Löschen, Bezahlen, Veröffentlichen oder Ablehnen nie ungeprüft an ein einzelnes Urteil hängen.

- ☐ Erfolg und Fehler sind an gelabelten Beispielen messbar.
- ☐ Unbekannte Fälle haben einen sicheren Ausweg.
- ☐ API-Kosten und Datenweitergabe sind bewusst geprüft.
- ☐ Ein Mensch kann Entscheidung und Modellversion nachvollziehen.

Jev ersetzt nicht dein KI-Abo. Es ergänzt Chatmodelle dort, wo Software oft dieselbe begrenzte Entscheidung treffen muss.

Kein vollständiger API-Produkttest. Preise, Limits und Early Access können sich ändern. Mehr KI-Anleitungen: @willi_neubauer.ai.

[Dokumentation öffnen](#) • Quellen: S2 • S4 • S5