
Start je bedrijfs- kennisbank voor Ai in 1 dag

Geef je Ai een wiki, geen zoekmachine. De LLM-wiki-aanpak: één feit op één plek, in gewone tekst, klaar in een dag.

Rudy Jellesma

rudyjellesma.nl

Werkdocument · 2026

■ Waarom dit kader?

De grootste versneller van Ai in je bedrijf is niet een slimmer model, maar een goede kennisbank. Een Ai die jouw manier van werken, jouw afspraken en jouw eerder genomen beslissingen kent, geeft antwoorden die kloppen. Een Ai zonder die context raadt.

De meeste bedrijven proberen dat op te lossen met een berg documenten waar de Ai elke keer opnieuw doorheen zoekt. Dat werkt matig: het is traag, het vindt vaak net het verkeerde stuk, en het herhaalt bij elke vraag hetzelfde zoekwerk. Er is een betere manier, en die kun je in één werkdag opzetten.

Geef je Ai een wiki, geen zoekmachine. Compileer je kennis één keer goed, in plaats van hem elke keer opnieuw te laten vinden.

Dit stappenplan beschrijft de LLM-wiki-aanpak, een idee dat Ai-onderzoeker Andrej Karpathy publiek deelde en dat ik zelf dagelijks gebruik als geheugen voor mijn Ai-agents. Het is verrassend low-tech: gewone tekstbestanden, geen dure software.

■ De vier principes

Een LLM-wiki draait om vier simpele regels. Houd je eraan en je kennisbank blijft bruikbaar; laat ze los en hij verzandt binnen een maand in een moeras.

- 1. Eén feit, één plek.** Elk gegeven staat op precies één plaats. Geen dezelfde afspraak in drie documenten, want dan raken die drie na verloop van tijd met elkaar in tegenspraak en weet niemand nog welke klopt.
- 2. Atomaire notities.** Kleine, gefocuste bestanden, één onderwerp per stuk. Een handige vuistregel: houd elke notitie klein genoeg om in één oogopslag te lezen, in de orde van hooguit een pagina of anderhalf. Een reus van een document waarin alles staat, is voor een Ai (en voor een mens) juist moeilijker te gebruiken dan tien scherpe notities.
- 3. Gewone tekst met onderlinge verwijzingen.** Schrijf in platte tekst (markdown), niet in een dichtgetimmerd bestandsformaat van één leverancier. Verbind notities met elkaar via verwijzingen, zodat de Ai van het ene onderwerp naar het verwante kan springen. Zo bouw je een web van kennis in plaats van een stapel losse blaadjes.
- 4. De Ai schrijft de wiki, niet andersom.** De ruwe input (vergadernotities, losse aantekeningen) laat je de Ai verwerken tot nette notities. De wiki is het compilaat, niet het archief. Antwoorden stapelen zich zo op: wat één keer is uitgezocht, staat er de volgende keer gewoon.

■ De drie lagen

Een gezonde kennisbank heeft drie duidelijk gescheiden lagen. Verwar je ze, dan ontstaat rommel.

Laag	Wat het is	Wie schrijft
Ruwe bronnen	Onbewerkte notities, clips, vergaderverslagen. Blijven ongewijzigd.	De mens (invoer)
De wiki	Nette, atomaire notities met verwijzingen. Het bruikbare kennislaag.	De Ai (compileert)
De inrichting	De afspraken over hoe een notitie eruitziet en hoe invoer verwerkt wordt.	De mens (config)

De ruwe bronnen gooi je nooit weg en pas je nooit aan; het zijn je originelen. De wiki is wat de Ai dagelijks gebruikt en bijhoudt. De inrichting is de handleiding die de Ai vertelt hoe hij dat moet doen.

■ Wat er in een gezonde kennisbank staat

Je hoeft niet met een leeg vel te beginnen. Een bruikbare LLM-wiki bestaat uit een handvol herkenbare soorten notities:

- **Conceptnotities:** uitleg van één onderwerp of werkwijze.
- **Entiteitnotities:** één per klant, project, leverancier of apparaat.
- **Samenvattingen:** het geheel op één pagina.
- **Vergelijkingstabellen:** opties naast elkaar (zoals de tabellen in dit document).
- **Vraag-en-antwoord-logs:** eerder beantwoorde vragen, zodat je ze niet twee keer stelt.
- **Een index:** de inhoudsopgave die de weg wijst.
- **Een doorlopend logboek:** waar je alleen aan toevoegt, nooit in wist.

■ Geen vector-database nodig (nog niet)

Een veelgehoorde reflex is dat je meteen dure zoektechnologie (een vector-database) nodig hebt. Dat is voor een beginnende kennisbank overbodig. Zolang je onder de honderd notities zit, past het geheel simpelweg binnen wat een Ai in één keer kan overzien. Een index plus gewone bestanden is genoeg. Bouw die zware machinerie pas als je er echt tegenaan loopt, niet uit voorzorg.

Dat de aanpak zich terugbetaalt, merk je snel. Toen ik in mijn eigen kennisbank de dubbelingen opruimde (dezelfde regels die in zeven bestanden herhaald stonden), scheelde dat ongeveer 900 tokens per sessie. Minder herhaling betekent minder kosten en, belangrijker, minder kans op tegenstrijdigheden.

■ De valkuil: laat een mens de promotie-poort bewaken

Nu de belangrijkste waarschuwing, want hier gaat het bij automatisering vaak mis. Het is verleidelijk om een Ai elke sessie automatisch lessen te laten wegschrijven naar je kennisbank. Doe dat niet zomaar.

Toen ik dat volledig automatisch liet lopen, was de overgrote meerderheid van wat er binnenkwam ruis: bijna-duplicaten, halve inzichten, dingen die niet klopten. Een goedkope automatische sorteerder produceert nu eenmaal veel bijna-hetzelfde. De kennisbank vervulde sneller dan hij groeide.

De oplossing is een tweetraps-aanpak met een mens als poortwachter:

1. **Staging-bak.** Nieuwe inzichten komen eerst in een wachtruimte terecht, niet meteen in de wiki.
2. **Promoveren na een patroon.** Pas als hetzelfde inzicht zich twee keer voordoet, promoveer je het tot een echte, vaste notitie of regel. Eenmalige toevalligheden filter je zo vanzelf weg.

De mens blijft de promotie-poort. De Ai mag voorstellen en verzamelen; wat blijvend in de kennisbank komt, bepaal jij. Nog twee praktische regels die veel ellende voorkomen: laat de Ai een notitie altijd vers herlezen vóór hij hem wijzigt (anders overschrijft hij een verse versie met een oude), en voer een wijziging overal door waar het feit voorkomt (anders sluipt tegenspraak binnen).

Eén harde grens tot slot: zet nooit wachtwoorden, sleutels of andere geheimen in je kennisbank. Een Ai-agent die je notities vrij mag lezen, leest die geheimen ook. Een audit in mijn eigen opzet vond ooit een echte, actieve sleutel die door automatisering in de kennisbank was beland. Geheimen horen in een kluis, niet in je wiki.

■ Het stappenplan voor één dag

Trek er één werkdag voor uit. Aan het eind heb je een werkende kennisbank die morgen al meewerkt.

Ochtend, deel 1 (circa 1 uur): kies je fundament. Maak één centrale map in gewone tekstbestanden. Bepaal je vier of vijf soorten notities (concept, klant/project, samenvatting, index, logboek). Schrijf de index als eerste, ook al is hij nog leeg.

Ochtend, deel 2 (circa 2 uur): leg je belangrijkste kennis vast. Begin met de tien tot vijftien dingen die je het vaakst uitlegt: je werkwijze, je vaste afspraken, je belangrijkste klanten of projecten. Eén onderwerp per bestand, kort en scherp. Laat de Ai ruwe aantekeningen voor je omzetten naar nette notities; jij controleert.

Middag, deel 1 (circa 2 uur): leg de verbindingen en de inrichting. Verbind verwante notities met verwijzingen. Schrijf de inrichting: een korte handleiding voor de Ai over hoe een notitie eruitziet, waar nieuwe input heen gaat, en de gouden regel "één feit, één plek".

Middag, deel 2 (circa 1 uur): richt de staging-bak in. Maak de wachtruimte voor nieuwe inzichten en spreek je promotie-regel af (twee keer gezien, dan pas vast). Test het geheel met een paar echte vragen aan de Ai en kijk of hij de juiste notities gebruikt.

Afsluiting (circa 1 uur): controleer en ruim op. Loop je notities na op dubbelingen (één feit, één plek), controleer dat er geen geheimen in staan, en werk de index bij. Klaar.

■ Tot slot

Een kennisbank voor Ai is geen groot IT-project. Het is een dag werk, een handvol principes en de discipline om een mens de poort te laten bewaken. De winst is blijvend: elke vraag die je goed beantwoordt, blijft beantwoord, en je Ai wordt met de week bruikbaar omdat hij jouw bedrijf steeds beter kent.

Begin klein: één map, vier soorten notities, één dag. De rest groeit vanzelf mee, zolang je bij elke toevoeging vasthoudt aan die ene regel: één feit, één plek.

Meer praktijkverhalen over Ai in het MKB, inclusief de kennisaanpak die dit allemaal aandrijft, lees je op rudyjellesma.nl. Vragen over dit stappenplan of hulp bij het opzetten voor jouw bedrijf? Ook daarvoor kun je daar terecht.

© Rudy Jellesma, rudyjellesma.nl. Dit document mag je vrij delen binnen je eigen organisatie.

■ Over de auteur

Rudy Jellesma is ondernemer en CTO. Hij bouwt en beheert AI-systemen die dag en nacht meewerken in zijn eigen bedrijven, van monitoring en codecontrole tot boekhouding en mail-afhandeling. Op rudyjellesma.nl deelt hij wat daarbij werkt en wat misgaat, steeds vertaald naar de praktijk van het MKB.

Verder lezen

Praktijkverhalen, tips en stappenplannen over AI voor het MKB vind je op rudyjellesma.nl.

Andere gratis downloads:

- Het Ai-stoplicht
- Ai-startklaar-checklist voor MKB
- Ai-kostenwijzer voor MKB
- BTW-checklist voor Ai-abonnementen
- Ai-boekhouden met akkoord-gate
- Beveilig je Ai-agent in 7 stappen

© 2026 Rudy Jellesma, rudyjellesma.nl. Dit document mag je vrij delen binnen je eigen organisatie.