Coaching-advies GreenFox AI-assistent

Beste Renzo,

Dank voor de heldere opdracht. Ik heb je mail gecombineerd met wat Hamza en Marcella de afgelopen weken hebben verteld, onderzoek gedaan naar wie dit soort dingen al heeft geprobeerd, en het geheel daarna nog eens kritisch tegen het licht gehouden voordat ik het naar jullie stuurde. Dat laatste is belangrijk: het eerste advies dat ik had, was te enthousiast. Deze versie is dat bewust minder.

Wat mij het meest opvalt: de echte drijfveer achter dit project is niet gemak, maar risicospreiding. Marcella zei het zelf letterlijk: een groot deel van jullie kernproces hangt aan één persoon, en als zijn vliegtuig neerstort heeft GreenFox een probleem. Dat besef stuurt mijn advies. Maar precies daarom moet ik het ook eerlijk brengen: dit wordt geen slimme bot die indruk maakt. Het moet een gecontroleerd werksysteem worden, met duidelijk eigenaarschap, meetbare doelen en harde grenzen. Zonder dat is het over een jaar weer een nieuw éénpersoonsrisico, alleen dan verstopt in een systeem in plaats van in een hoofd.

Eerst het bewijs, en de eerlijke grens daarvan

Bestaat dit al ergens? Kort antwoord: gedeeltelijk, en de losse onderdelen wel, het geheel niet. Ik heb dit uitgezocht en wil het ook eerlijk brengen: de cijfers hieronder zijn leverancierscijfers en case-schaal die niet één-op-één naar GreenFox vertaalt. Het zijn dus aannemelijke aanwijzingen, geen bewijs dat het bij jullie ook zo gaat.

- **The Adecco Group + Salesforce Agentforce**: AI-agents die per recruiter het voorscreenen overnemen. 1,2 miljoen kandidaat-interacties in 10 landen, time-to-deliver -50 procent. Andere schaal en ander proces dan GreenFox, maar het laat zien dat dit soort automatisering in de branche werkt.
- **IDR + Bullhorn Copilot**: het dichtst bij wat jullie willen — generatieve AI ín het systeem per recruiter, met conceptmails in eigen toon. Hun COO noemt het "hun eigen assistent naast zich, elke dag."
- **Moderna + OpenAI**: geen staffing-bedrijf, maar wél het bewijs voor "per rol, niet één generieke bot": 750-plus maatwerk-assistenten, 100 procent adoptie op de juridische afdeling.

- **IBM AskHR**: de maatstaf voor interne kennisbots. 94 procent van de vragen zonder mens opgelost, 40 procent lagere HR-kosten in vier jaar.
- **Malt + Dust**: bewijst dat meerdere kleine, gespecialiseerde assistenten beter werken dan één generieke bot.
- **Lexi AI (Nederland)**: een CAO-kennisbot voor de uitzendbranche, dichtst bij jullie contractbewaking, maar zonder onafhankelijk bevestigde cijfers.

En dan de eerlijke keerzijde: het complete concept dat jullie voor ogen hebben — kennisbot, ochtendbriefing, conceptmails en contractbewaking, allemaal per medewerker persoonlijk — bestaat nergens als geheel. Dat kan betekenen dat jullie voorlopen. Het kan ook betekenen dat het te complex, te duur of te lastig beheerbaar is om in zijn geheel te bouwen, en dat niemand het daarom heeft afgemaakt. Ik geloof het eerste, maar de manier om dat te bewijzen is klein beginnen, niet in één keer alles bouwen.

Volledig onderzoek: https://verslag.haartips.nl/r-greenfox-succesverhalen-fable.html

---

a. De technische opzet op jullie eigen server

Bouw het als een interne webapplicatie binnen jullie Microsoft-omgeving, geen app-store-app. Hamza beheert Intune volledig en kan telefoons met één druk resetten, dus een webapp achter jullie eigen inlog is veilig bereikbaar op elke laptop en Fairphone die jullie al beheren.

Over de hardware: ik had eerder stellig geadviseerd om voor Apple-hardware met unified memory te kiezen boven een RTX-videokaart. Dat trek ik terug tot een aanname. Wat het beste werkt hangt af van welk model je draait, hoeveel mensen tegelijk vragen stellen en wie het beheert. Laat Hamza eerst een kleine benchmark draaien met de twee kandidaten en dezelfde testvragen, en beslis daarna. Een voorkeur is geen feit.

Werk in twee sporen voor het model zelf. Spoor één: bouw en test snel met een API van een grote aanbieder, in een EU-regio, zonder dat er ooit kandidaat- of klantgegevens in gaan. Spoor twee: draai parallel een lokaal of Europees model (Mistral is het bekijken waard) op de eigen server voor het gevoelige werk, zodat je nooit van één Amerikaanse partij afhankelijk bent.

De architectuur: één centrale motor, met daaromheen niet meteen persoonlijke bots per medewerker, maar eerst twee gedeelde assistenten. Dat is een bewuste bijstelling van mijn eerste advies. Begin met een kennisassistent (vraag-en-antwoord op het Proceshuis) en een

schrijfassistent (conceptmails, nog niet per persoon). Pas als beide bewezen waarde leveren, ga je personaliseren per medewerker. Dat voorkomt dat je complexiteit bouwt voordat je weet of de basis werkt.

---

b. Het Proceshuis structureren zodat de bot betrouwbaar antwoordt

Zet het Proceshuis om naar kleine, adresseerbare blokken: één pagina per processtap, met eigenaar, systeem, termijn, versie en datum. De vijftien kernregels staan apart, als harde laag boven alles, nooit afhankelijk van of de bot toevallig het juiste stukje tekst terugvond. Alle getallen en termijnen in tabellen, niet in lopende tekst.

Citaatplicht en scope-weigering blijven de kern: alleen antwoorden uit de kennisbasis, altijd met bron, en eerlijk "dit weet ik niet" als het antwoord er niet in staat. Maar dat is niet genoeg. Voeg een derde regel toe: wanneer stuurt de bot iemand door naar een mens in plaats van te weigeren of te gokken? Bijvoorbeeld bij tegenstrijdige regels, bij een vraag die net buiten de letterlijke tekst valt maar wel binnen het onderwerp, of bij twijfel over een bedrag of termijn. Zonder die regel loopt de bot vast tussen "ik weet het" en "ik weet het niet", terwijl de praktijk vaak grijs is.

Regel ook governance: wie mag de kennisbasis wijzigen, en wie is eigenaar van de waarheid als het Proceshuis, de CAO, de praktijk van een ervaren collega en een teamlead het onderling niet eens zijn? Zonder een aangewezen eigenaar per proces krijg je een bot die netjes klinkt maar op termijn verouderde of tegenstrijdige antwoorden geeft.

Bouw een testset vóórdat je vertrouwt, en breder dan alleen goed-fout: veertig vragen met het juiste antwoord en bron, tien strikvragen waarvan het antwoord bewust ontbreekt, en daarnaast een kleine set grensgevallen (vragen die net anders geformuleerd zijn dan de brontekst) en conflictvragen (twee bijna-tegenstrijdige regels tegelijk). Herhaal deze testset bij elke wijziging, als een vaste regressietest, niet als eenmalige keuring. Leg bij elke versie van kennis, prompts en profielen vast welke versie actief was toen een antwoord werd gegeven — dat is je enige manier om achteraf te kunnen reconstrueren wat er misging als een antwoord niet klopte.

c. Het gedeelde en later persoonlijke profiel

Begin, zoals bij a genoemd, met één GreenFox-brede stijl en toon voor de schrijfassistent, niet meteen met een profiel per medewerker. Als je daarna gaat personaliseren: bouw dat met echte voorbeeldmails, niet met een beschrijving. Maar let op een risico dat ik eerder niet noemde: stijl en inhoud zijn niet hetzelfde. Als je alleen op toon traint, kan de assistent ook iemands minder goede gewoontes overnemen, bijvoorbeeld te lange mails of het vermijden van lastige boodschappen. Beoordeel daarom niet alleen "klinkt dit als deze medewerker", maar ook apart "is dit ook gewoon een goede mail".

En een tweede risico, rechtstreeks uit het gesprek met Marcella: medewerkers voelen zich snel gecontroleerd. Spreek daarom niet alleen af, maar bouw ook technisch af, dat een assistent nooit gebruiksgegevens per persoon naar een leidinggevende stuurt. Een afspraak is een intentie, een technische beperking is een garantie. Beide heb je nodig, want ook een teamlead met de beste bedoelingen kan dit anders gaan gebruiken zodra het kan.

---

d. De bouwvolgorde, met meetpunten en een stopcriterium

De volgorde blijft grotendeels hetzelfde, met twee toevoegingen die ontbraken: harde meetpunten, en een moment om te stoppen als het niet werkt.

Eerst de kennisbasis en de bredere testset. Dan de kennisvragen-functie met citaatplicht en de escalatieregel. Daarna, niet per medewerker maar gedeeld, de schrijfassistent voor conceptmails. Dan pas, als beide functies aantoonbaar gebruikt worden, de contract- en subsidiebewaking als apart traject: begin daar met alleen signaleren, read-only, gezien de zakelijke waarde en het risico dat daar samenkomen. Personaliseren per medewerker komt als laatste, niet als eerste.

Meet vanaf dag één, niet alleen "gebruikt hij het uit zichzelf" maar concreet: tijdwinst per taak, hoeveel conceptmails zonder aanpassing worden verstuurd, hoe vaak de assistent terecht escaleert naar een mens, en hoe medewerkers het vertrouwen beoordelen. En spreek vooraf een stopcriterium af: bijvoorbeeld, als na zes weken minder dan een vooraf afgesproken percentage van de antwoorden bruikbaar is, of de assistent nauwelijks gebruikt wordt, stoppen jullie of bouwen jullie het opnieuw, in plaats van door te blijven investeren in iets dat niet aanslaat.

Fase 2, het meeluisteren via Plaud, blijft geparkeerd tot fase 1 bewezen is.

AVG, iets steviger dan "geen persoonsgegevens erin"

Dat is een goed startpunt maar niet voldoende. Leg voor de bouw begint ook vast: wie is verwerker en wie verwerkingsverantwoordelijke voor dit specifieke kanaal, welke logging jullie aanhouden en hoe lang die bewaard blijft, of de gekozen modelaanbieder op verzoek geen trainingsdata van jullie prompts gebruikt, wat de doelbinding is van elk stukje kennis dat je deelt, en of er subverwerkers zijn waar je vooraf toestemming voor nodig hebt. Dit hoeft geen zwaar traject te zijn, maar wel een bewuste keuze per punt, gedocumenteerd, in plaats van een aanname.

Een levend voorbeeld

Ik bouw een werkend demo-kennisbot, met verzonnen voorbeelddata in de structuur van het Proceshuis, waar je zelf een vraag kan intypen en een echt antwoord terugkrijgt, inclusief bron of een eerlijke weigering. Zodra die staat, krijg je een link. Dit is een illustratie van de werking, geen kant-en-klaar product: Hamza bouwt de echte versie na op jullie eigen server met echte data.

Mijn eigen rol, en waar die stopt

Ik adviseer, ik bouw niet. Om te zorgen dat dit ook zonder mij blijft werken, hoort daar wat mij betreft bij: dat Hamza bij elke stap een korte beheerhandleiding bijhoudt naast de code, dat er een eenvoudig incidentproces is voor als de assistent een keer duidelijk fout antwoordt, en een fallback: wat doet een medewerker als de assistent een dag niet beschikbaar is. Dat hoort niet bij de techniek, maar bepaalt of dit iets wordt waar GreenFox zelfstandig op kan bouwen.

Concrete eerste stap voor komende week

Van jullie kant: Hamza stuurt de Proceshuis-hoofdstukken Plaatsen en Ontwikkelen plus de vijftien kernregels, alleen methodiek, en zet een testmailbox met fictieve gegevens klaar. Spreek met elkaar af wie namens GreenFox eigenaar is van welk kennisonderdeel, en prik het vaste wekelijkse moment voor de coaching-vraag.

Van mijn kant: binnen een week na ontvangst lever ik de demo-kennisbot op, met het uitgebreide testset-format (inclusief grensgevallen en conflictvragen), zodat Hamza die daarna zelf onderhoudt.

Groet,
Jamal