

Van prompt naar prototype in 7 minuten

23 maart 2026, 15:16

Mihail Mihaylov
Wim Codenie

Hoe een genAI-coderingsassistent een forecasting-tool bouwde

In een vorig artikel toonden we hoe foundationmodellen op basis van tijdreeksen forecasting voor industriële bedrijven veel eenvoudiger maakt. Die genAI-modellen kunnen betrouwbare voorspellingen doen zonder training, tuning of specialistische expertise. Ze zijn open source en draaien op een standaardlaptop. De traditionele kostendrempel voor nauwkeurige voorspellingen verdwijnt dus snel.

Het vervolg van het verhaal

Forecastingmodellen zijn echter maar een deel van de uitdaging.

Zelfs als het model op zich klaar is voor gebruik, moet iemand nog steeds de omringende software bouwen. Gegevens moeten worden geladen, het model moet worden aangeroepen, resultaten moeten worden verwerkt en gevisualiseerd, en randgevallen moeten worden behandeld.

Traditioneel waren hiervoor ontwikkelaars nodig. Bij veel projecten waren er zelfs hele softwareteams nodig om forecasting-pipelines en -applicaties te bouwen.

Dat roept een praktische vraag op.

Als genAI de kosten van forecastingmodellen heeft verlaagd, kan het dan ook de kosten verlagen van het bouwen van de software eromheen?

GenAI-codeerassistenten

Een nieuwe generatie tools geeft stilaan een antwoord op die vraag.

Codeerassistenten zoals Claude Code en Codex kunnen rechtstreeks software genereren uit instructies in natuurlijke taal. In plaats van handmatig code te schrijven, beschrijven gebruikers wat ze willen bouwen. De assistent genereert de aanvraag dan in een mum van tijd.

De systemen doen meer dan het automatisch aanvullen van code. Ze kunnen projecten structureren, afhankelijkheden beheren en de architectuur aanpassen naarmate de eisen veranderen.

Ze vervangen geen ontwikkelaars, maar ze verkorten de weg van idee tot werkend prototype wel drastisch.

Voor bedrijven die met forecasting willen beginnen, verandert dat het proces. In plaats van zowel datawetenschappers als software-engineers nodig te hebben, kan één domeinexpert met de juiste tools vaak al een eerste proof-of-concept bouwen.

Wat we hebben gebouwd en in hoeveel tijd

Bij Sirris hebben we die verschuiving aan den lijve ondervonden.

Drie maanden lang hebben we Claude Code gebruikt om een uitgebreide forecasting-simulator te bouwen. Het was niet zomaar een projectje. De simulator ondersteunt verschillende foundationmodellen op basis van tijdreeksen, zowel lokaal als in de cloud, samen met verschillende datasets, prestatiemetingen, configureerbare parameters, visualisaties en exportfuncties.

3 maanden werk. Een serieuze onderzoekstool.

Vervolgens voerden we een experiment uit.

We vroegen Claude Code om de simulator te analyseren en een specificatie op te stellen voor een heel eenvoudige versie: een minimale proof-of-concept forecaster die dagelijks een Excel-bestand met historische tijdreeksgegevens inleest en een voorspelling voor de volgende dag genereert, met behulp van het best presterende foundationmodel uit onze

benchmarks.

Claude maakte de specificatie.

Toen grepen we opzettelijk in. We openden een compleet nieuwe sessie met de assistent. Zonder kennis van de simulator. Een schone lei.

We gaven de specificatie in een prompt en vroegen om de forecaster te bouwen.

Zeven minuten later hadden we een werkend prototype.

Geen kladversie. Geen pseudocode. Een complete applicatie die gegevens laadt, het model aanroept en voorspellingen produceert. Met één extra prompt kan ze zelfs worden voorbereid voor integratie in een productiepijplijn.

Wat 3 maanden duurde, was plots in 7 minuten gedaan?

Voor de duidelijkheid: het prototype van zeven minuten is niet gelijk aan de simulator van drie maanden. De simulator is een uitgebreid onderzoeksgereedschap met veel functies. Het prototype is met opzet minimaal en richt zich op één taak.

Maar dat is precies het punt. De meeste bedrijven die forecasts willen beginnen maken, hebben geen volledig onderzoeksplatform nodig, maar een proof-of-concept dat een eenvoudige vraag beantwoordt: werkt dit voor onze gegevens?

Tot voor kort vergde het beantwoorden van die vraag weken van ontwikkeling en een speciaal team. Nu kan het vaak in een paar minuten.

De drie maanden werk waren niet voor niets geweest. Ze gaven de specificatie meer info: welk model moet er worden gebruikt, hoe moet de pijplijn worden gestructureerd en welke parameters zijn van belang? Domeinexpertise is nog steeds bepalend voor de kwaliteit van het resultaat. De codeerassistent neemt gewoon het knelpunt van de implementatie weg.

Wat betekent dit in de praktijk?

Neem het typische traject van een industrieel bedrijf dat forecasting onderzoekt. Eerst moet het management de investering goedkeuren. Vervolgens analyseert een data science-team de gegevens, selecteert het modelleerbenaderingen en bouwt het pipelines. Er kunnen maanden voorbijgaan voordat het eerste resultaat er is.

We bekijken nu het alternatief.

Een domeinexpert met toegang tot een codeerassistent beschrijft het probleem. Binnen enkele minuten of uren draait er een werkend prototype op hun machine. Als de resultaten er veelbelovend uitzien, kan de oplossing worden verfijnd. Zo niet, dan heeft het experiment een middag gekost in plaats van een kwartaal.

Bij de overgang naar productie is nog steeds goede engineering vereist, maar de verkenningsfase gaat van maanden naar dagen. Bedrijven kunnen verschillende forecastingideeën testen in plaats van zich vast te leggen op één groot project.


Range Slider

© Freepik

Er zijn twee dingen veranderd

Open source-foundationmodellen op basis van tijdreeksen leveren nu een voorspellingsnauwkeurigheid die vergelijkbaar is met die van dure industriële oplossingen, zonder training of expertise op het gebied van machinelearning. Daarnaast kunnen codeerassistenten een beschrijving in natuurlijke taal binnen enkele minuten omzetten in een werkend forecastingprototype.

Samen verlagen die ontwikkelingen de kosten van forecasting drastisch. Problemen die vroeger te duur waren om te onderzoeken, zoals het voorspellen van het energieverbruik van één gebouw of het voorspellen van de vraag naar nicheproducten, worden plotseling haalbaar.

Als je tijdreeksgegevens hebt, kun je veel eerder beginnen met forecasten dan je misschien zou denken.

Ontdek de volledige reeks

Dit artikel maakt deel uit van een driedelige reeks over Generatieve AI voor tijdreeksvoorspelling:

Intro: Uw activiteiten genereren data. Gebruikt u die al voor voorspellingen?

Artikel 1: Generatieve AI maakt voorspellen toegankelijk

Artikel 2: Van prompt tot prototype in 7 minuten

Ontdek het zelf in 3 uur

Neem deel aan de Sirris masterclass over GenAI voor tijdreeksvoorspelling. Geen programmeerkennis vereist. Geen voorafgaande ML-ervaring nodig.

U gaat naar huis met een praktische stapsgewijze aanpak die u meteen kunt toepassen op uw eigen data. Bijna gratis voor Vlaamse bedrijven dankzij de VLAIO-subsidie.

Schrijf u in voor de masterclass

Auteurs



Mihail Mihaylov



Wim Codenie