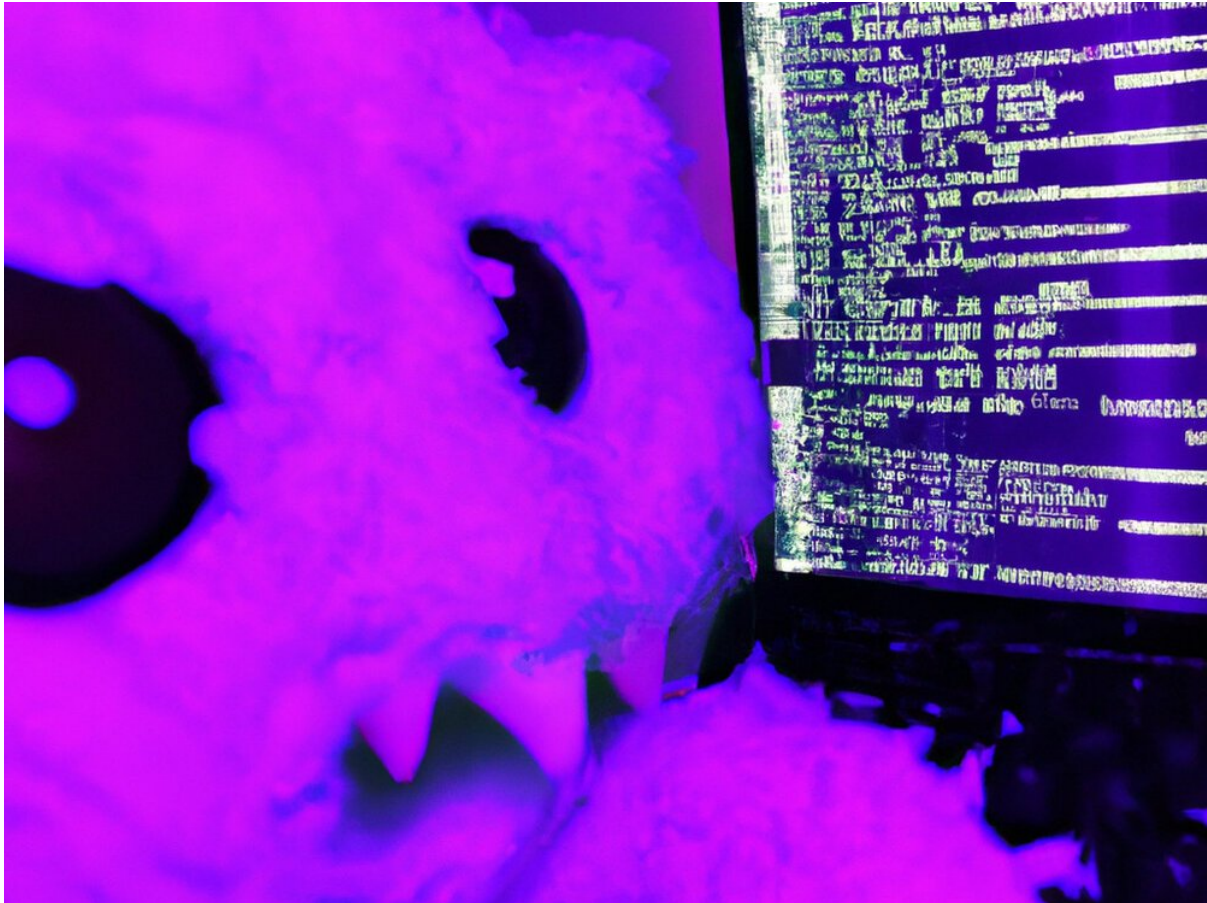


AI gebruiken om code te genereren

Programmeren met superkrachten

Willem L. Middelkoop

3 mei 2023



Als onderdeel van mijn software-optimalisatie om cloudkosten te verlagen, moest ik een bestaand stuk inefficiënte serversoftware vervangen door iets dat gebruik maakt van robuustere (maar gefragmenteerde) tools die beschikbaar zijn in Debian GNU/Linux. Kon het GPT4 taalmodel me wat AI-magie bieden? Lees verder!

Onderdeel van de clouddiensten van mijn bedrijf is de zogenaamde "Lemmid Manager", een geïntegreerd contentmanagementsysteem dat ruwe input (tekst, video's, afbeeldingen) neemt en mooie, goed presterende statische websites genereert. Geconfronteerd met [stijgende cloudkosten](#), ben ik bezig delen van de software te herschrijven om efficiënter te zijn: minder computerkracht gebruiken om dezelfde resultaten te bereiken.

Afbeelding rotatie

Je realiseert het je misschien niet, maar wanneer je je telefoon gebruikt om een foto te maken, wordt deze altijd liggend (of breedbeeld) opgeslagen. Wanneer je een staande (of rechtopstaande) foto maakt, voegt je telefoon simpelweg een beetje extra informatie toe die de manier waarop je de camera hebt gehouden opslaat - zodat wanneer je de foto terugkijkt, deze correct georiënteerd wordt weergegeven. Het is sneller om één beetje (extra) informatie te schrijven dan om de daadwerkelijke bits en bytes waaruit je afbeelding bestaat te roteren.

Dit is allemaal prima als de foto je telefoon niet verlaat, maar wanneer je de foto naar een website uploadt, moet deze geoptimaliseerd worden voor het web. De daadwerkelijke afbeeldingsbits en -bytes moeten dan de daadwerkelijke inhoud van de foto weerspiegelen. De afbeelding kan dan [verder geoptimaliseerd worden voor laadsnelheden](#) en verschillende viewports.

Automatische afbeeldingsrotatie op basis van EXIF

Moderne (smartphone) camera's slaan de oriëntatie op in de zogenaamde EXIF die de daadwerkelijke afbeeldingsgegevens vergezelt. Om de afbeelding (bits en bytes) correct de oriëntatie van de camera te laten weerspiegelen, moet men de EXIF-gegevens lezen en op basis van de oriëntatie een rotatie-algoritme toepassen op de afbeeldingsbits en -bytes.

Huidige implementatie

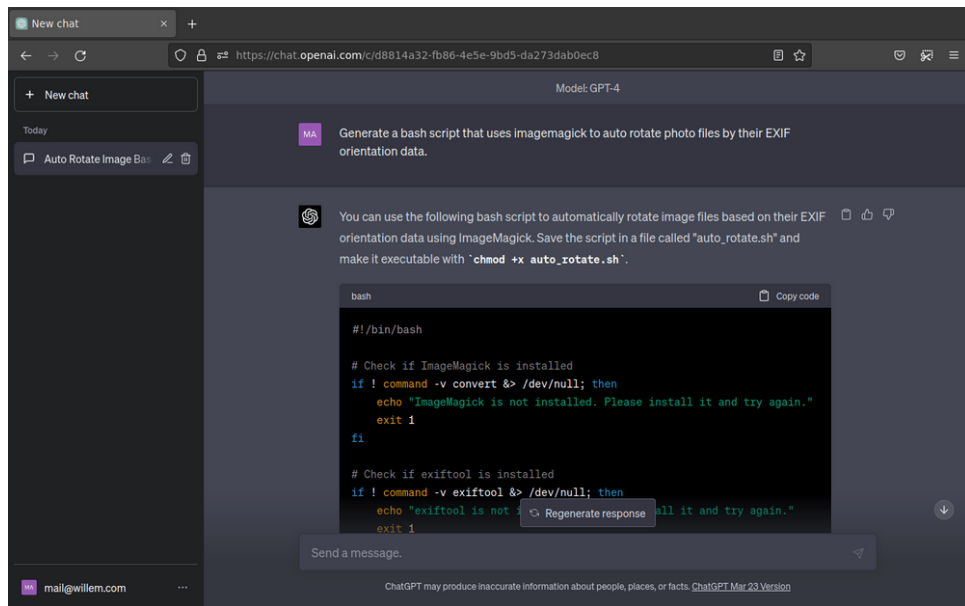
De originele (handgeschreven) implementatie gebruikt NodeJS en een module genaamd "lqip" (LightWeight ImageProcessor) om dit te doen. Dit werkt prima in de meeste situaties, maar naarmate afbeeldingen groter zijn geworden (vanwege meer megapixels), zijn de geheugenvereisten aanzienlijk toegenomen. Het nadeel van alles in één NodeJS-proces te doen, is dat het geheel een soort geheugenvreter wordt - waardoor de volledige NodeJS-backend eronder lijdt wanneer je hem tot het uiterste drijft door duizenden foto's te verwerken.

Een veelgebruikte strategie om knelpunten in een stuk software te optimaliseren, is ze uit het hoofdprogramma te halen. In plaats van afbeeldingen rechtstreeks vanuit NodeJS te verwerken, wil ik veelgebruikte Debian GNU/Linux-tools (die hun sporen verdiend hebben) gebruiken om de automatische afbeeldingsrotatie uit te voeren. Vanuit NodeJS kan ik dan de externe 'auto_rotator' aanroepen en wachten tot deze zijn taak heeft voltooid. Door een NodeJS-proces te laten wachten op een asynchroon resultaat/callback, maak je het weer beschikbaar voor andere verzoeken.

GPT4 als codegenerator

Het mooie van vrije software is dat er veel bouwstenen beschikbaar zijn die je kunt gebruiken om je eigen programma's te bouwen. Ik wist dat Debian GNU/Linux tools beschikbaar heeft die de afbeeldingsrotatie op basis van EXIF-gegevens kunnen uitvoeren. De uitdaging is om één script te maken dat deze verschillende tools gebruikt om de klus te klaren en tegelijkertijd enige robuustheid biedt bij het afhandelen van uitzonderingen.

In plaats van zelf diep in alle beschikbare commando's en commando-opties te duiken, besloot ik GPT4 te vragen om een script voor me te genereren dat precies doet wat ik wilde, met behulp van standaardtools die beschikbaar zijn in Debian GNU/Linux:



Code genereren is net zo makkelijk als een beschrijvende prompt intypen

Binnen enkele seconden genereerde OpenAI's GPT4 een bash-script dat [ImageMagick](#) (een veelgebruikte grafische tool) gebruikt om aanpassingen uit te voeren op basis van de EXIF-gegevens in een bepaald afbeeldingsbestand. De AI selecteerde de "exiftool" om de oriëntatiegegevens te lezen en op basis van de output van die tool het "convert"-commando van ImageMagick uit te voeren om de juiste rotatie toe te passen.

```

1 #!/bin/bash
2
3 # Check if ImageMagick is installed
4 if ! command -v convert &> /dev/null; then
5     echo "ImageMagick is not installed. Please install it and try again."
6     exit 1
7 fi
8
9 # Check if exiftool is installed
10 if ! command -v exiftool &> /dev/null; then
11     echo "exiftool is not installed. Please install it and try again."
12     exit 1
13 fi
14
15 # Check if at least one file is provided
16 if [ "$#" -eq 0 ]; then
17     echo "Usage: ./auto_rotate.sh [file1] [file2] ..."
18     exit 1
19 fi
20
21 # Loop through all provided files
22 for file in "$@"; do
23     # Check if the file exists
24     if [ ! -f "$file" ]; then
25         echo "File '$file' does not exist."
26         continue
27     fi
28
29     # Extract the orientation using exiftool
30     orientation=$(exiftool -Orientation -n -s "$file")
31
32     # Perform rotation based on the orientation value
33     case "$orientation" in
34         1)
35             echo "Image '$file' has the correct orientation."
36             ;;
37         3)
38             echo "Rotating image '$file' 180 degrees."
39             convert "$file" -rotate 180 "$file"
40             ;;
41         6)
42             echo "Rotating image '$file' 90 degrees clockwise."
43             convert "$file" -rotate 90 "$file"
44             ;;
45         8)
46             echo "Rotating image '$file' 90 degrees counterclockwise."
47             convert "$file" -rotate -90 "$file"
48             ;;
49         *)
50             echo "Image '$file' has an unsupported orientation value. Skipping."
51             ;;
52     esac
53
54     # Reset the orientation to normal
55     exiftool -Orientation=1 -n -overwrite_original "$file"
56 done

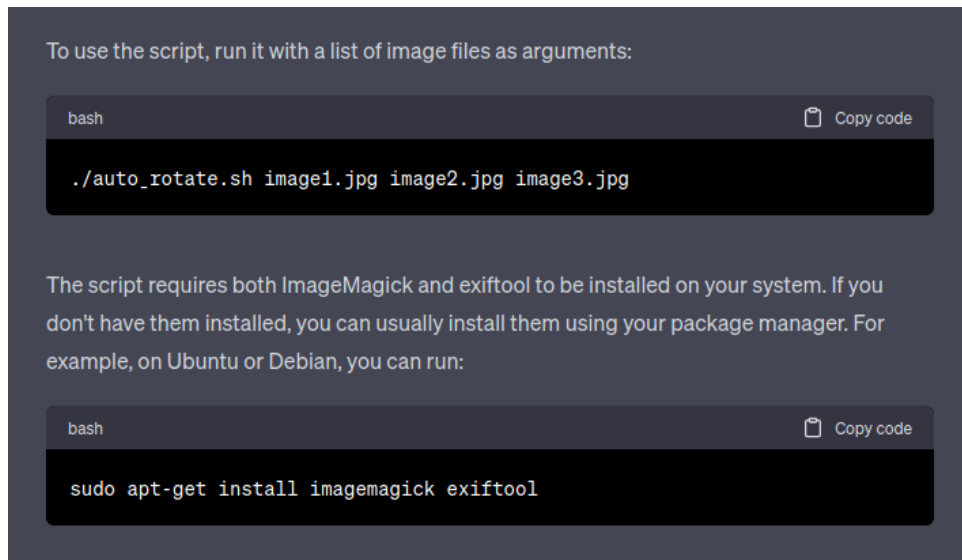
```

GPT4's gegenereerde output code "auto_rotate.sh" roteert afbeeldingen met ImageMagick op basis van EXIF-oriëntatie

De volledige code (die binnen enkele seconden werd gegenereerd) is prachtig commentariseerd, waardoor je een goed gevoel krijgt van wat er gebeurt. Het is gestructureerd en geoptimaliseerd om door mensen te worden gelezen.

Documentatie

Naast de daadwerkelijk werkende code leverde de AI documentatie waarin werd uitgelegd hoe de nieuw gegenereerde software moet worden gebruikt - inclusief installatie-instructies. Dit is iets waarvan ik weet dat de meeste programmeurs het over het hoofd zien (technische documentatie is vaak een bijzaak).



AI's instructies voor het gebruik van de gegenereerde code - de meeste echte programmeurs kunnen leren van de helderheid in communicatie

Conclusie

Binnen een paar minuten had ik een werkend, gedocumenteerd stuk code dat me helpt een knelpunt in een stuk cloudsoftware op te lossen. Dat is gewoon geweldig - het bespaarde me tijd (die ik vervolgens heb gebruikt om deze blogpost te schrijven).

Je kunt je alleen maar afvragen welke effecten grote taalmodellen zoals ChatGPT/GPT4 zullen hebben op hoe we werken. Ik omarm AI-tools volledig - weten hoe je ze moet gebruiken zal in de nabije toekomst essentieel zijn (denk ik)!



Een programmeur uit de toekomst - zoals verbeeld door DALL-E van OpenAI