KOFETKE.

Kaj jezikovni model v resnici počne

Zakaj lahko sistem reši zahtevno nalogo in si vseeno izmisli preprosto dejstvo?



MODELI IN ZANESLJIVOST

Razumljiv pregled žetonov, učenja, konteksta, orodij in preverjanja.

SLOVENSKA IZDAJA

september 2026
Praktična uporaba AI

Viri preverjeni 21. septembra 2026.

Odgovor jezikovnega modela je lahko videti, kot da ga je napisal človek, ki je prebral vsako datoteko, razumel položaj in preveril računanje. Včasih je uporaben osnutek. Včasih je tekoče zapisana izmišljotina. Razlika je le redko v domiselnem navodilu modelu. Izhaja iz razumevanja, kaj sistem prejme, kaj lahko pregleda, kaj sme storiti in kako nekdo preveri rezultat.

Jezikovni model besedilo obdeluje kot žetone: majhne dele besedila, ki niso nujno cele besede. Isti stavek lahko v različnih sistemih zasede različno število žetonov. Število žetonov vpliva na to, ali zahteva ustreza omejitvi, koliko stane zahteva API-ja in koliko časa traja. OpenAI žetone opisuje kot dele, ki jih jezikovni modeli berejo in pišejo, pretvorbo angleških besed pa navaja le kot grobo orientacijsko pravilo.[1] Slovenščina, imena, URL-ji, tabele in mešani jeziki lahko od angleške ocene precej odstopajo. Kadar so pomembni stroški ali zmogljivost, je zato dokument treba izmeriti v dejanski aplikaciji.

Kaj se napoveduje

Učenje in uporaba sta ločeni dejavnosti. Med učenjem razvijalci s pomočjo velikih zbirk primerov prilagajajo številne številske parametre. Med uporabo, ki se običajno imenuje sklepanje, že naučeni sistem prejme trenutni vhod in ustvari nadaljevanje ali drug izhod. Običajno ne išče po spletu, ne pregleduje službenega diska in se ne spomni preteklega pogovora, razen če mu aplikacija te stvari posreduje. Naučeni model je ena komponenta, ne popoln pomočnik.

Za jezikovni model je neposredna naloga izbira verjetnih naslednjih žetonov iz besedila, ki ga je prejel. Ta opis je uporaben, vendar je preveč skop, da bi pojasnil vsak sistem umetne inteligence (AI). Generatorji slik delajo s slikovnimi predstavitvami in pogosto z besedilnim pogojevanjem. Govorni sistemi lahko zvok pretvorijo v besedilo ali ustvarjajo zvok. Sistemi zaznavanja lahko razvrščajo ali poiščejo značilnosti na sliki. Nekatere aplikacije združujejo več modelov s podatkovnimi zbirkami, pravili in človeškim pregledom. Če vse to imenujemo napovedovalna tipkovnica, prikrijemo razlike, ki so v praksi pomembne.

Kontekstno okno omejuje, koliko gradiva lahko jezikovni model med izmenjavo obravnava. Navodila, priloženo ali pridobljeno gradivo, zgodovina pogovora in rezultati orodij skupaj z odgovorom, ki nastaja, zasedajo to omejitev. Anthropic ga opredeljuje kot besedilo, na katero se model lahko sklicuje med ustvarjanjem odgovora, vključno z odgovorom samim, in ga razlikuje od učnih podatkov.[2] Bolje ga je razumeti kot začasno delovno gradivo kot pa kot kartotečno omaro.

Veliko kontekstno okno lahko vsebuje daljši priročnik s pravili, vendar zmogljivost ni razumevanje. Dolgi vnosi lahko zakopljejo odločilni stavek, vključijo nasprotujoče si različice pravilnika ali porabijo omejitev, ki bi jo bilo bolje nameniti čistemu izseku. Anthropic opozarja, da se lahko z naraščanjem števila žetonov zmanjšata natančnost in priklic, ta učinek pa imenuje »context rot«.[2] Trenutno pogodbeno določilo, vrstico računa ali preglednico odločitev postavite blizu navodila; ne nalagajte nefiltriranega arhiva v klepet in temu ne recite pridobivanje informacij.

Plasti, ki jih pogosto zamenjujemo

Spodnji izrazi ločujejo naloge, ki jih trženjske spletne strani pogosto združujejo.

Plast	Preprost pomen	Vprašanje, ki si ga velja zastaviti
Model	Naučena komponenta, ki vhod pretvori v izhod	Katera točna različica je ustvarila ta rezultat?
Uteži	Naučeni številski parametri, ki jih uporablja model	Ali jih lahko prenesemo in zaženemo ter pod katero licenco?
Aplikacija	Zaslon za klepet ali delovni tok okoli modela	Do katerih datotek, orodij in ljudi lahko dostopa?
Pridobivanje informacij	Izbira ustreznih dokumentov ob uporabi	Iz katerega vira, s katerega datuma in s katerim dovoljenjem je nastal ta izsek?
Orodje	Nadzorovano dejanje, denimo iskanje po podatkovni zbirki ali ustvarjanje osnutka	Kaj lahko spremeni in kdo ga odobri?
Vrednotenje	Ponovljiv preizkus na podlagi določenih primerov	Kaj pri tej nalogi šteje kot sprejemljiv rezultat?

Javno dostopne uteži tudi niso sopomenka za odprtokodnost. Opredelitev Open Source Initiative zahteva uporabne informacije o podatkih, kodo in parametre za prednostno obliko spreminjanja; uteži opisuje kot naučene parametre, ki so povezani z arhitekturo modela.[4] Ponudnik lahko dovoli prenos uteži, hkrati pa zadrži učne podatke, uporablja pravilnik o uporabi ali zahteva navedbo avtorstva. To je lahko še vedno koristno. Preprosto potrebuje natančnejšo oznako: uteži, ki jih je mogoče prenesti, model z javno dostopnimi utežmi ali imenovana licenca, namesto domneve o neomejenih pravicah.

Vmesnik za programiranje aplikacij oziroma API omogoča programski opremi, da od drugega sistema zahteva določeno operacijo. API gostovanega modela del odgovornosti prenese na dobavitelja. Podjetje pošilja zahteve na končno točko in plačuje po pogojih tega ponudnika. OpenAI navaja, da se podatki API-ja ne uporabljajo za učenje ali izboljšanje njegovih modelov, razen če stranka izrecno privoli, njegova dokumentacija pa hkrati opisuje dnevnike spremljanja zlorab, ki se lahko privzeto hranijo do 30 dni.[5] To sta dve različni dejstvi. »Ne uporablja se za učenje« ne pomeni »nikoli se ne hrani«. Ponudnikova potrošniška klepetalna storitev, API, regionalna uvedba v oblaku in integracija tretje osebe imajo lahko različne pogoje.

Izmišljen delovni tok: preverjanje ponudb

Razmislite o izmišljenem ljubljanskem oblikovalskem studiu North Window. Pošilja ponudbe s fiksnim cenikom, projektnim koledarjem in občasnimi prilagojenimi izjemami. Lastnik želi, da model pred pošiljanjem ponudbe označi tvegano besedilo. Sistem prejme osnutek, trenutni cenik in kratek seznam odobrenih

določil. Lahko opozori na manjkajoči stavek o avansu ali datum, ki ni skladen s koledarjem. Ne more poslati ponudbe, spremeniti cenika, obljubiti pravnega določila ali izmisliti popusta.

Prva različica je odpovedala na prepričljiv način. Pretekla ponudba v mapi za pridobivanje informacij je vsebovala poseben popust za nekdanjo stranko. Model je posnemal njen ton in predlagal popust, ne da bi opazil, da je potekel. Besedilo je zvenelo samozavestno, ker se je naučil, kako ponudbe navadno zvenijo. Ni vedel, da ukinjena izjema ne pomeni pooblastila.

Popravek je bil večinoma običajno upravljanje informacij. Studio je shranil eno datoteko s trenutnim cenikom in datumom veljavnosti, stare ponudbe premaknil v arhiv zunaj pridobivanja informacij, zahteval, da vsako opozorilo navede vrstico vira, ki jo je uporabilo, in lastniku pustil vlogo pošiljatelja. Preverjanje po pravilih je pred prikazom nasveta modela o besedilu primerjalo valuto, odstotek avansa in datume. Model je postal pregledovalec omejenega osnutka, ne odločevalec.

Pridobivanje informacij lahko zagotovi nove poslovne podatke, vendar navodilo, zapisano v pridobljeni datoteki, ne dobi pristojnosti nad aplikacijo. Članek 03 pojasnjuje nadzor nad različicami dokumentov, dostopom in besedilom, podobnim navodilom.

Zakaj lahko prepričljiv odgovor odpove

NIST uporablja izraz »konfabulacija« za ustvarjeno vsebino, ki je napačna ali neresnična, vendar predstavljena samozavestno. Njegov profil za generativno AI navaja, da lahko statistično

napovedovanje ustvari natančne in dosledne izhode ter izhode, ki so netočni ali notranje nedosledni.[3] To je lastnost, ki jo je treba obvladovati, ne odstotek, ki ga lahko preberemo s kartice modela.

Obstaja več pogostih načinov odpovedi. Model lahko napačno razume dvoumno zahtevo. Lahko uporabi star pridobljeni dokument. Pri nalogi z več pogojnimi koraki lahko slabo računa. Lahko navede vir, ki ne podpira njegove trditve. Lahko sledi zlonamernemu navodilu, vdelanemu v uvoženo vsebino. Aplikacija, ki uporablja orodja, lahko izvede napačno dejanje, ker je bil nepravilno izbran identifikator. Posodobitev modela lahko spremeni izhod, tudi če navodilo ostane enako. Dokumentacija OpenAI navaja, da odgovori v klepetu privzeto niso deterministični, zato se ponovljene zahteve lahko razlikujejo.[1]

Odgovor ni zahteva, naj bo model »natančen«. Odgovor je, da delo spremenimo v preverljiv postopek. Za natančno določene aritmetične izračune, podvojitve, datume, davčne izračune in preverjanje dovoljenj uporabite deterministično programsko opremo. Za aktualno gradivo z navedljivim virom uporabite pridobivanje informacij. Model uporabite za razvrščanje, povzemanje, pripravo osnutkov ali razlago negotovosti. Osebi pustite pristojnost za zaveze, ki vključujejo denar, pogodbe, zaposlovanje, zdravstvene informacije, regulirane nasvete ali obljube strankam.

Sklepanje, orodja in dovoljenja

Nekateri izdelki opisujejo model kot takega, ki zna sklepati. To je lahko uporabna oznaka za model, nastavljen tako, da pri zahtevnih korakih porabi več računanja, ali za aplikacijo, ki ga prosi, naj nalogo razčleni. Ni pa dokaz, da je bil izhod neodvisno

preverjen. Vidna razlaga vmesnih korakov je lahko sama po sebi napačna, nepopolna ali prilagojena tako, da zveni prepričljivo. Preizkus ostaja, ali se končna trditev ujema z izvorno evidenco, izračunom ali odobrenim pravilom.

Orodja tveganje spremenijo bolj kot bolj zgovoren odgovor. Orodje za iskanje samo za branje lahko pridobi zapis stranke; orodje za e-pošto lahko ustvari zunanjo zavezo; plačilno orodje lahko premakne denar. Vsako orodje potrebuje ozko omejeno dovoljenje, zanesljiv identifikator, evidenco revizije in jasno točko odobritve. Pomočnik bi moral znati povedati, katero dejanje predlaga, zakaj in katere informacije je uporabil. Kjer človek dejanja ne more zlahka razveljaviti, se mora sistem ustaviti pred izvedbo.

Uporabna delitev dela je, da model predlaga, programska oprema preveri, odgovorna oseba pa odobri pomembne zaveze. To podaljša čas pregleda. Ali to skupaj prihrani delo, je odvisno od pogostosti napak sistema in stroška njihovega popravka; vrednotenje mora šteti oboje.

Vrednotenje pred zaupanjem

Uporabno vrednotenje se začne z ozko nalogo. North Window bi lahko sestavil preskusni nabor 30 anonimiziranih preteklih osnutkov ponudb: običajni primeri, zastarel popust, manjkajoči avans, nasprotujoč si datum začetka, dvojezična opomba in datoteka, ki vsebuje stavek, podoben navodilu. Lastnik pri vsakem primeru zabeleži pričakovano opozorilo, izvorno datoteko

in sprejemljiv odgovor. Sistem se preizkuša z isto različico modela, nastavitvami pridobivanja informacij in orodji, ki bodo uporabljeni v produkciji.

Preglednica rezultatov naj meri napake, ki studio stanejo čas ali denar:

Preverjanje	Pogoj za uspeh	Odziv ob napaki
Povezanost z virom	Vsako opozorilo se poveže s trenutno odobreno datoteko	Opozorila ne prikažite in zabeležite manjkajoči vir
Varnost cen	Brez neodobrenega predloga cene ali popusta	Zahtevajte pregled lastnika; popravite pridobivanje informacij ali pravila
Skladnost datumov	Označen je spor s koledarjem	Primerjajte z evidenco koledarja, ne z besedilom modela
Vzdržanje odgovora	Nejasen primer je označen za pregled	Ne zahtevajte samozavestnega odgovora
Ponovljivost	Ponovljeni preskusni zagoni ostanejo znotraj dogovorjenega odstopanja	Raziščite spremembe navodila, modela ali ponudnika

NIST priporoča pragove zmogljivosti ali zahtevano raven zagotovil o ustreznem delovanju sistema kot del odobritve uvedbe, njegova navodila o življenjskem ciklu pa preskušanje, vrednotenje,

preverjanje in validacijo obravnavajo kot stalno delo.[3][12] Za majhno podjetje je to lahko preglednica, mapa z različicami preskusov in mesečni pregled vzorcev. Ne potrebuje raziskovalnega laboratorija. Potrebuje imenovane primere, pravilo za uspeh, shranjene izhode in lastnika, ki lahko ustavi delovni tok.

Zanesljivost pripada celotni ureditvi: vhodom, dovoljenjem, pridobivanju informacij, modelu, navodilu, orodjem, preverjanjem in človeški odobritvi. Model je lahko osupljiv, pa je še vedno neprimeren za nalogo brez preverjanja. Praktično vprašanje je ožje: katera omejena naloga ustvari rezultat, ki ga je mogoče pregledati, preden vpliva na koga drugega?

Model naj za naslednjo osebo pusti dovolj dokazov, da lahko rekonstruira odločitev: različico vhodnih podatkov, pridobljene datoteke, izhod, zahteve orodij, preverjanja in odobritev. Ta evidenca prepričljiv pogovor spremeni v delo, ki ga je mogoče kritično presoditi, popraviti in ponoviti.

Viri

Številke virov sledijo skupnemu seznamu. Povezave vodijo do izvirnih dokumentov. Pogoji uporabe in razpoložljivost storitev se lahko spremenijo.

[1] OpenAI: advanced text-generation usage
https://developers.openai.com/api/docs/guides/advanced-usage

[2] Anthropic: context windows
https://docs.anthropic.com/en/docs/build-with-claude/context-windows

[3] NIST: Generative AI Profile
https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.600-1.pdf

[4] OSI: Open Source AI Definition
https://opensource.org/ai/open-source-ai-definition

[5] OpenAI: API data controls
https://developers.openai.com/api/docs/guides/your-data

[12] NIST: AI actor tasks across the lifecycle
https://airc.nist.gov/airmf-resources/airmf/appendices/app-a-descriptions-of-ai-actor-tasks