Automatisch nakijken

Met behulp van Sonja van Dam heb ik gekeken naar mogelijkheden om met een LLM automatisch na te kijken. De tussentijdse conclusie is dat er potentie is, maar dat het antwoordmodel geschikt voor mensen suboptimaal is als antwoordmodel voor een LLM. Ik doe een voorstel voor een vervolg experiment om antwoordmodellen aan te scherpen met behulp van LLM.

Introductie

Een LLM is met name sterk in het herkennen van patronen in syntax en semantiek. Omdat mensen intelligentie vaak afleiden via taal wekt een LLM de indruk intelligent te zijn. In de praktijk valt dat vaak tegen. Zo zijn LLM's tot nu toe slecht in schaken[1]. Vaardigheden zoals basale wiskunde worden wel correct uitgevoerd, maar de methode om tot de uitkomst te komen is twijfelachtig[2]. Ook lopen modellen nog ver achter op gezond verstand, zelfs als deze modellen een PhD denkniveau worden toebedeeld door de makers[3]. In tegenstelling tot velen denk ik niet dat LLM's waardeloos zijn als ze (nog?) niet intelligent zijn. Een uitstekende taal beheersing kan heel nuttig zijn, mits het probleem dat we willen oplossen geformuleerd kan worden als een puur taal probleem. Automatisch examens en testen nakijken is een probleem, dat naar mijn idee geformuleerd kan worden als een taak, waarbij taalbeheersing afdoende is om de taak naar wens af te ronden.

Opzet van het experiment

Ik heb van Sonja van Dam een tentamen "sustainable impact" mogen inzien, samen met het antwoordmodel en heb de pseudonieme antwoorden en scores gekregen van ongeveer 250 studenten. In eerste instantie heb ik naar de eerste 5 vragen gekeken. Enerzijds om het handwerk aan mijn kant te beperken, want het antwoordmodel was niet gemakkelijk foutloos automatisch in te laden in een geschikt data formaat. Anderzijds om de kosten van LLM gebruik te beperken tijdens deze verkennende fase.

Na het inladen in een database van: de vragen, het antwoordmodel voor deze vragen en alle antwoorden van studenten op de vragen; heb ik voor elk student antwoord aan verschillende LLM's gevraagd in hoeverre dat antwoord voldeed aan de criteria in het antwoord model. Hierbij kregen de LLM's de volgende opties:

- Aangeven welke criteria toepasbaar zijn
- Aangeven welke criteria misschien toepasbaar zijn
- Op basis van de vraag, antwoordmodel en het antwoord een aantal vrije opmerkingen maken

Op basis van de toegekende criteria heeft een computer programma vervolgens scores berekend en deze scores zijn vergeleken met de scores, die studenten in werkelijkheid hebben gekregen. Het werk van de LLM kan op drie manieren geevalueerd worden:

- De mate waarin het de LLM lukt om (met zekerheid) de gegeven criteria te koppelen aan de antwoorden. Dit geeft aan hoeveel werk een LLM uit handen kan nemen.
- De mate waarin de LLM afwijkt van de werkelijke toegekende scores door een mens. Ik heb gekozen voor ICC2[4] om de mate van overeenstemming vast te stellen. Verder heb ik de MAE berekent om een indicatie te krijgen hoeveel scores afwijken[5].

1 https://www.theregister.com/2024/06/04/chess_puzzle_benchmark_llm/
2 https://www.anthropic.com/research/tracing-thoughts-language-model#mental-math
3 https://simple-bench.com/
4 https://en.wikipedia.org/wiki/Intraclass_correlation#Use_in_assessing_conformity_among_observers
5 https://en.wikipedia.org/wiki/Mean_absolute_error

- De bias van een LLM is simpelweg (LLM-score – docent-toekenning) / maximale punten * 100. Dat betekent dat bij een negatieve percentage de docent meer punten toekende en bij een positief percentage de LLM meer punten toekende. Deze bias berekening dient ter indicatie wat er verwacht mag worden aan verandering in de cijfers voor studenten.

Een nadeel van deze experiment opzet is dat uit de uiteindelijke student scores moeilijk het toegepaste criterium van een docent is af te lezen. Als er bijvoorbeeld twee criteria zijn met twee punten, dan is aan een score van 2 punten niet af te lezen welk criterium is toegepast.

Het is mijn hypothese dat LLM's in staat zijn goed te presenteren op bovenstaande vlakken. Niet omdat ze een diep begrip hebben van de inhoud, maar omdat de taak is gereduceerd tot een simpele vergelijking van syntax en sematiek patronen tussen twee teksten A) het antwoord van de student en B) het opgestelde antwoord model. Merk op dat ik alle berekeningen aan scores uitvoer in software en niet overlaat aan LLM's.

Resultaten

In totaal zijn er drie LLM's met wisselende configuraties getest. De besteede tijd aan het experiment is voornamelijk gaan zitten in evaluatie en verbetering van resultaten. Hierdoor ben ik niet toegekomen aan verschillende aanbieders proberen. Alle modellen zijn hierdoor van OpenAI. Per model geef ik aan hoe goed het model de taak heeft uitgevoerd, hoeveel werk het bespaart zou hebben en wat de kosten zijn om de taak uit te voeren.

O1 reasoning
Ten tijde van het experiment was dit het beste "reasoning" model[6]. De resultaten zijn teleurstellend. Vanwege de lange interne monoloog, die een model uitvoert als onderdeel van de "reasoning" ligt het gebruik van tokens hoog. Er wordt afgerekent per token en slechts na ongeveer 40 studenten werd mijn ingestelde limiet van €100 bereikt. Geschatte kosten liggen daarmee op €625 euro per tentamen.

De MAE ligt tussen de 0.4 en 1.4 met een gemiddelde van 0.8. De bias is 0.9%, dus die is te verwaarlozen. Er waren helaas onvoldoende observaties om significante ICC2 waardes te berekenen. Ik ben snel doorgegaan naar goedkopere alternatieven.

O1-mini (3n voted)
Met O1-mini blijven we bij het "reasoning" paradigma, maar werken we met een model dat beperktere capaciteiten heeft. Het wordt voornamelijk gebruikt om mee te programmeren en is heel goedkoop. Het is naar mijn weten nog onduidelijk wat precies de effecten zijn van de mengeling tussen natuurlijke taal en programmeertaal syntax en semantiek. In het kader van dit experiment zijn wel positieve effecten te verwachten, omdat het antwoord uiteindelijk uitgelezen moet worden door software en daarbij helpt een voorkeur voor rigoreuze software syntax. Met "3n voted" wordt bedoeld dat de vraag 3x is gesteld en het meest gegeven antwoord wordt gekozen als uiteindelijke uitkomst.

Dit model is in staat om ongeveer 39% van de studenten te beoordelen zonder tussenkomst van een docent. De kwaliteit van deze oordelen is een stuk beter dan bij O1. De MAE ligt tussen de 0.2 en 0.9 met een gemiddelde van 0.5. De bias loopt hier iets op naar 4,9%, maar blijft in een redelijke marge. Er is helaas slechte correlatie tussen de scores van de LLM en de docenten, die is onvoldoende bij alle vragen. Het grote voordeel van dit model zijn de kosten. Zelfs met 3n zijn de geschatte kosten voor het tentamen €6,50.

6 https://platform.openai.com/docs/guides/reasoning?api-mode=chat

GPT4.1 (3n voted)
Net als met O3-mini wordt met dit model de vraag 3x is gesteld en het meest gegeven antwoord wordt gekozen als uiteindelijke uitkomst. Het basis model is echter geen reasoning model, maar is speciaal gemaakt voor gebruikers die modellen in combinatie willen gebruiken met andere software, zoals bijvoorbeeld in dit experiment om op grote schaal vragen te stellen en te rekenen met de uitkomsten.

Van alle gesteste modellen werkt dit model het beste. Het ontziet de docenten voor ongeveer 2/3 van de antwoorden. De MAE ligt tussen de 0.3 en 0.7 met een gemiddelde van 0.5. De bias krimpt ten opzicht van O3-mini naar 1,2%. Maar belangrijker is dat er voor het eerst een redelijke tot goede overeenstemming is met significantie tussen docenten en de LLM voor 3 van de 5 vragen. Daarmee is er zeker ruimte voor verbetering, maar het resultaat kan mijns inziens niet worden afgedaan als kansloos. De kosten voor een tentamen nakijken zijn ongeveer €30. Een stuk duurder dus dan O1-mini, maar nog steeds veel goedkoper dan O1.

GPT4.1 (3n voted, human in the loop)
Hier is het model en manier van prompten niet veranderd ten opzichte van bovenstaande GPT4.1. Echter de manier van scores berekenen is anders om te simuleren dat een docent als extra controle naar het toegekende criterium heeft gekeken. Elke keer als een model aangeeft dat een criterium “misschien toepasbaar” is, dan wordt de score van dit criterium toegepast mits de score overeenkomt met dat van de docent. Dit is om resultaten te simuleren waarbij een docent een gedeelte van het werk van de LLM nakijkt waar deze “twijfelt”, door met een druk op de knop goed te keuren.

Deze methode van human in the loop is levert echter geen significante verbeteringen op. Het is daarom beter om geen gebruik te maken van een onderscheid tussen “toepasbaar” en “misschien toepasbaar” bij het vragen naar criteria.

GPT4.1 (3n voted, comments review)
Ook hier is het model en manier van prompten niet veranderd ten opzichte van bovenstaande GPT4.1. Het toekennen van scores werkt zo, dat wanneer een LLM een comment geeft, dat de correcte score van de docent wordt overgenomen. In dit geval simuleren we resultaten waarbij een docent de comments van een LLM doorkijkt en waar nodig de toegekende criteria verbetert.

Deze methode is wel veelbelovend. Ongeveer de helft van alle toegekende criteria komt met een opmerking van de LLM. Dus het werk dat blindelinks wordt overgenomen verminderd tot 1/3. Echter de overeenstemming tussen docent en LLM neemt drastisch toe. De overeenstemming groeit naar goed tot uitstekend in 4 van de 5 vragen. De MAE komt tussen 0.1 en 0.4 te liggen met een gemiddelde van 0.3. Alleen de bias neemt in negatieve zin toe naar 2,5%, maar blijft dus redelijk.

Discussie en volgende stappen

Ik heb voor de verschillende GPT4.1modellen gekeken naar waar mogelijke verbeterpunten liggen. Vooral bij vraag 4 valt op dat het model consequent een ander criterium kiest dan de docenten, als er niet aan comments review wordt gedaan. Het model zou drastisch verbeteren als het lukt de LLM iets vaker mee te laten stemmen met de docent, die deze vraag heeft nagekeken (elke vraag is door een enkele docent beoordeeld). Als je kijkt naar de opmerkingen bij toegepaste criteria van vraag 4, dan valt op dat de LLM consequent oordeelt dat er “onvoldoende terminologie” wordt gebruikt. Dit is typisch het soort fout dat verwacht mag worden van LLM’s volgens mij. Puur op basis van syntax en semantiek is namelijk geen oordeel te vellen over wat “voldoende” is. Mijn vermoeden is dan

ook, dat als we het antwoordmodel kunnen aanpassen en explicieter maken, dat de LLM dan beter de taak zal uitvoeren en handmatige checks van de opmerkingen minder nodig zullen zijn.

Een mogelijke richting is om een LLM te laten reflecteren op het gegeven antwoord model en verbeteringen voor te stellen. Dit zou eventueel wel een duurder “reasoning” model kunnen zijn, waarbij een mens de suggesties naloopt en goedkeurt. Het voordeel van deze methode is dat inladen van een antwoordmodel wel nog grotendeels automatisch kan gaan en dat het docenten niet veel werk hoeft te kosten. Bovendien is er de mogelijkheid om de vragen zelf specifieker te maken, wanneer deze nog niet is voorgelegd aan studenten, en dat kan de kwaliteit van nakijken ten goede komen. Een mogelijk nadeel van deze methode is dat docenten met weinig prompt ervaring, toch nog fouten kunnen maken, die pas tijdens het nakijken zelf zullen blijken.

Een andere richting is om docenten te laten reageren op opmerkingen waar de LLM mee komt. De opmerkingen kunnen dan gebruikt worden om het antwoordmodel te verbeteren. Vervolgens kunnen alle antwoorden opnieuw door de LLM gehaald worden, met gebruik van het verbeterde antwoord model. Dit proces kan zich herhalen totdat er geen opmerkingen meer zijn. Deze iteratieve methode heeft als voordeel, dat de docent continue betrokken is en goed zicht heeft op de gevolgen van aanpassingen aan het antwoordmodel. Een nadeel is dat deze methode relatief duur is. Kosten kunnen beperkt worden door opmerkingen eerst te laten clusteren. Dit clusteren is vrij goedkoop en kan er toe leiden, dat docenten met relatief weinig opmerkingen doornemen, toch veel verbeteringen in één keer kunnen aanbrengen. Hoe minder iteraties er nodig zijn om het antwoordmodel te verbeteren hoe goedkoper dit proces wordt.

Conclusies

Het gebruik van LLM als tool bij het nakijken van tentamens lijkt kansrijk. Met relatief weinig aanpassingen aan het tentamen kan zo een tool ongeveer 1/3 van het werk uit handen nemen, zonder veel op de kwaliteit in te boeten. Vervolg experimenten zouden zich kunnen richten op hoe een LLM gebruikt kan worden om het antwoordmodel aan te scherpen voor gebruik bij automatisch nakijken. Dit proces van aanscherpen kan vooraf of achteraf het tentamen plaatsvinden, maar vraagt in alle gevallen een zekere betrokkenheid van docenten. Daarnaast kan het huidige experiment nog uitgebreidt worden met meer verschillende LLM aanbieders. Met name Antropic en Google hebben interessante modellen op de markt, die het proberen waard zijn in de toekomst.