Primjena metode mješavine stručnjaka u detekciji disruptivnih otkrića i inovacija

Zlatar, Fran

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1590

PRIMJENA METODE MJEŠAVINE STRUČNJAKA U DETEKCIJI DISRUPTIVNIH OTKRIĆA I INOVACIJA

Fran Zlatar

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1590

PRIMJENA METODE MJEŠAVINE STRUČNJAKA U DETEKCIJI DISRUPTIVNIH OTKRIĆA I INOVACIJA

Fran Zlatar

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

ZAVRŠNI ZADATAK br. 1590

Pristupnik:	**Fran Zlatar (0036542367)**
Studij:	Elektrotehnika i informacijska tehnologija i Računarstvo
Modul:	Računarstvo
Mentor:	doc. dr. sc. Mario Brčić

Zadatak: **Primjena metode mješavine stručnjaka u detekciji disruptivnih otkrića i inovacija**

Opis zadatka:

Veliki jezični modeli (engl. Large Language Models, LLM) su se pokazali izvanrednim u dohvatu informacija u različitim modalnostima sadržaja, te sve više napreduju u odgovaranju na pitanja i zaključivanju. No, postoje situacije sa manje egzaktnim pitanjima i odlukama gdje korištenje jednog modela izlaže korisnika riziku. Detekcija disruptivnih otkrića i inovacija je svakako jedna takva domena. Metoda mješavine stručnjaka (engl. Mixture of Experts, MoE) LLM-ova koristi skup više modela za povećanje kvalitete odluka sa nižim troškovima izvedbe, veću prilagodljivost različitim vrstama podataka i potencijalno bolju detekciju suptilnih signala koji bi inače mogli proći nezamijećeno kod metoda koje koriste samo jedan model. Cilj ovog završnog rada jest istražiti primjenu metode MoE u detekciji disruptivnih otkrića kao anomalija u znanstvenim člancima. Treba razviti sustav koji će omogućiti precizno prepoznavanje signala ili važnih informacija unutar teksta znanstvenih radova za investitore u pojedina područja interesa. Dodatno, treba usporediti MoE sa alternativom korištenja jednog, znatno većeg, monolitnog LLM-a u navedenoj zadaći. Implicirana dodatna zadaća jest složiti prikladan skup podataka koje korišteni LLM-ovi nisu vidjeli u tom formatu tijekom svog treninga.

Rok za predaju rada: 14. lipnja 2024.

Posebne zahvale profesoru i mentoru doc. dr. Mariu Brčiću na izdvojenom vremenu, trudu i znanju koje je uložio i prenio tijekom izrade ovog rada.

Sadržaj

Uvod

Posljednjih godina razvoj umjetne inteligencije te posebno velikih jezičnih modela (engl. Large Language Models, LLMs) i interes za te teme postao je sve veći. No s razvojem takvih modela i tehnologija dolaze mnogi tehnološki i praktični problemi. Jedan od takvih problema je kako napraviti sustav koji će imati što više neurona u svojoj neuronskoj mreži, a uz to znati kada se koji neuroni moraju „paliti“. Takva podjela zadataka unutar sustava koji pokušava replicirati ljudski mozak tojest funkcionirati na sličan način je vrlo izazovna i za sobom povlači dodatne tehnološke probleme. Cilj inženjera umjetne inteligencije (engl. Artificial Intelligence, AI) i strojnog učenja (engl. Machine Learning, ML) je dizajnirati i napraviti sustav koji bi imao karakteristike ljudskog mozga. Glavne karakteristike bi bile brzina odgovora, ali prije svega i točnost tog odgovora. Navedene karakteristike nije ni približno jednostavno implementirati na velikim ili čak ogromnim sustavima klasičnim metodama. Klasičnim metodama brzina je obrnuto proporcionalna točnosti. Povećamo li broj neurona u našoj neuronskoj mreži koja nam predstavlja ljudski mozak u mnogim slučajevima dobit ćemo točnije odgovore, no brzina odgovora će se značajno smanjiti. Stoga se umjesto klasičnih metoda za navedenu problematiku primjenjuje metoda mješavine stručnjaka (engl. Mixture of Experts, MoE). Kao i u ljudskom mozgu, metoda mješavine stručnjaka pokušava razdijeliti našu neuronsku mrežu (mozak našeg sustava) na više dijelova od kojih je svaki zadužen za specifične zadatke. Takvim pristupom ubrzava se vrijeme odgovora našeg sustava te se povećava točnost odgovora za razliku od klasičnih metoda gdje se ona time smanjuje. Cilj ovog rada je istražiti metodu mješavine stručnjaka, primijeniti ju nad testnim podatcima te usporediti rezultate sa jednim monolitnim velikim jezičnim modelom (model bez metode mješovitih stručnjaka), koji ima sličan ukupan broj neurona, nad tim istim podatcima. Očekivani rezultat bi bio da primjenom metode mješavine stručnjaka dobijemo točnije i brže odgovore. Ideja ovog rada je primjena metode mješavine stručnjaka u detekciji disruptivnih otkrića kao anomalija u znanstvenim člancima. Drugim riječima tražimo one članke koji su potencijalno veliko otkriće (signal) ili novost koja će u narednim godinama igrati značajnu ulogu u životima nas ili neke specifične grupe ljudi (npr. investitori). Primjer jednog signala u tehnološkom svijetu bio bi razvoj kvantnog računala (engl. Quantum computer) visokih performansi koje može raditi na temperaturama bliskim 0°C. Takav primjer je signal jer kvantna računala trenutno mogu raditi jedino na temperaturama blizu apsolutne nule (0K). Pomoću pažljivo odabranih i obrađenih članaka

prikupljenih s raznih stranica na internetu skupili smo skup podataka za testiranje ove metode. U budućim godinama možemo očekivati sve veću primjenu metode mješavine stručnjaka na razne druge općenitije probleme. Glavni karakteristika nužna za ovakav sustav je visoka razina preciznosti budući da je osnovna primjena sustava u krugu investitora koji bi prema njegovim rezultatima morali moći donijeti odluku o investiciji u pojedino područje ili patent. No, unatoč tome što je primjena ovakvog sustava usko vezana uz investicije, ovaj sustav je moguće primijeniti i na druge grane djelatnosti uz male preinake. Za primjenu metode mješavine stručnjaka koristio sam model Mixtral 8x7B razvijen od strane francuske tvrtke Mistral AI, a za usporedbu sa monolitnim modelom koristio sam GPT-3.5-Turbo američke firme OpenAI.

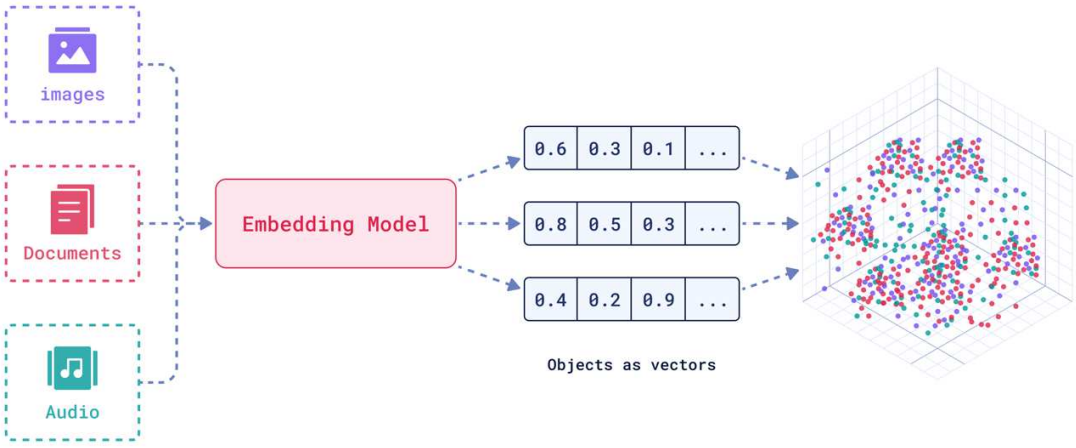
Teorijska podloga

Veliki jezični modeli (LLM)

Veliki jezični modeli (engl. Large Language Models, LLM) su veliki modeli dubokog učenja koji su trenirani, učeni, na velikim količinama tekstualnih podataka [1]. Da bi dobili dojam o veličini tih modela, GPT-3 model tvrtke OpenAI ima oko 300 miliona tokena. Token može biti jedna riječ ili dio riječi te ovisi o samoj konfiguraciji modela.

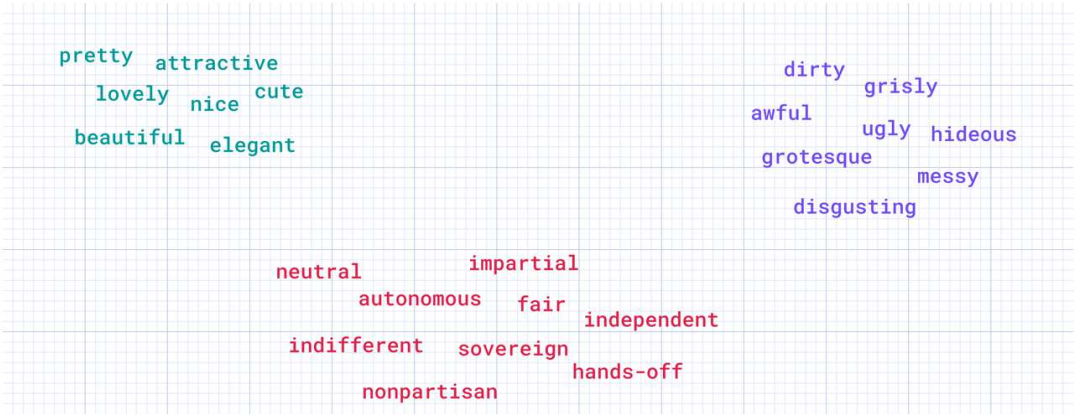
Kako rade LLM-ovi?

Veliki jezični modeli koriste arhitekturu transformatora. Transformator je vrsta neuronske mreže pogodna za obradu sekvencijskih podataka poput teksta. Rade na temelju promatranja međusobnih odnosa između samih neurona pomoću matematičkih modela, posebno vektora s realnim vrijednostima (engl. embedding). Ti vektori su vrlo velike dimenzije te njihova veličina ovisi o modelu. Oni predstavljaju točku u n-dimenzijonalnom prostoru, gdje je n veličina vektora ili vektorskog prostora [2].



Slika 1. Pojednostavljen shematski prikaz načina rada vektora u velikim jezičnim modelima.[2]

Kada tekst dođe na ulaz velikog jezičnog modela prvo prolazi kroz fazu pretvaranja teksta u tokene gdje se tekst rastavlja na manje jedinice, odnosno tokene, koji će onda biti zapisani u obliku vektora. Svaki takav token će imati svoje mjesto u tom vektorskom prostoru, kao što je prikazano na slici 1, te će se matematičkim alatima moći pratiti međusobni odnosi između tokena. Tokeni koji se češće pojavljuju skupa bit će bliže unutar tog prostora, dok će oni koji se rjeđe pojavljuju skupa biti udaljeniji, što je vidljivo na slici 2.



Slika 2. Prikaz sličnih vektora u vektorskom prostoru.[2]

Zašto koristimo vektore?

Tradicionalne relacijske baze podataka nisu efikasne kada imamo ogromne količine podataka. Potreban je bolji mehanizam kojim ćemo predstaviti odnose između jedinki tojest tokena. Takav problem rješavaju vektorske baze podataka koje pohranjuju prikaz riječi kao vektore. Postoje jednostavni matematički alati koji nam omogućavaju da izračunamo sličnost ili udaljenost između dva vektora. Neki od njih su:

- Sličnost kosinusa

$$\cos\theta = \frac{\boldsymbol{A}\cdot\boldsymbol{B}}{\|\boldsymbol{A}\|\|\boldsymbol{B}\|} \quad \{1\}$$

- Euklidska udaljenost

$$d(\boldsymbol{A},\boldsymbol{B}) = \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2} \quad \{2\}$$

- Manhattanska udaljenost (L1 udaljenost)

$$d(\boldsymbol{A},\boldsymbol{B}) = \sum_{i=1}^{n}|A_i - B_i| \quad \{3\}$$

Upotreba

Veliki jezični modeli pokazali su se odličnim za rješavanje zadataka poput obrade prirodnog jezika. ChatBot-ovi poprimaju sve veću pažnju zbog LLM-ova. Sve više kompanija koristi ChatBot-ove kako bi automatizirali proces pružanja usluge svojim korisnicima. Sposobni su pronaći ključne informacije među velikom količinom podataka i teksta. Automatski prevodioci teksta bolji su nego ikada zahvaljujući LLM-ovima. Moguće ih je primijeniti i u medicinske svrhe na razne načine i na razna područja. Jedno takvo područje je prevođenje

teksta u govor ili obrnuto za gluhonijeme osobe. Sve više se radi na tome, ali već i sada su veoma sposobni te mogu generirati programski kod ili čak cijele aplikacije. Mnogi smatraju da bi sve veća primjena ovakvih sustava mogla uzrokovati gubitke poslova jer će potreba za radnom snagom biti sve manja. S time dolaze i etička pitanja korištenja i razvijanja ovakvih sustava.

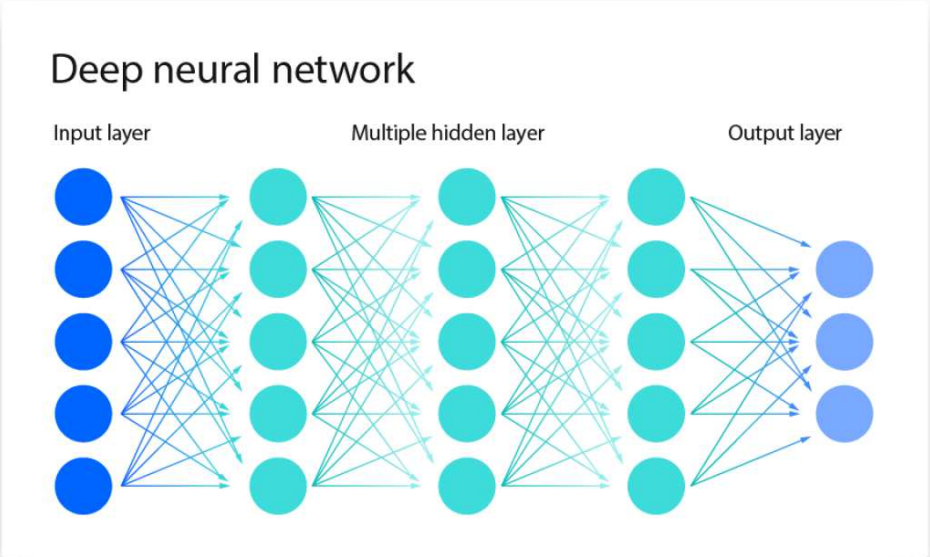
Etička pitanja

Sve većom primjenom velikih jezičnih modela raste strah populacije od gubitka poslova i sve intenzivnijeg razvijanja takvih sustava u smjeru gotovo potpune zamjene čovjeka za tehnologiju. Navedena problematika ne odnosi se samo na programere i slična zanimanja već i na cijelu populaciju. Vrlo lako se može primijetiti kako već sada velik broj multinacionalnih kompanija koristi velike jezične modele u obliku ChatBot-ova koji zamjenjuju dotadašnju radnu snagu u području korisničke podrške. Školski primjer je korisnička usluga u aviokompanijama. Još jedan veliki problem predstavlja intelektualno vlasništvo proizvoda koje veliki jezični modeli izrađuju. Veliki jezični modeli sposobni su generirati slike koje se mogu koristiti u razne svrhe kao što su na primjer logotipi, ali i umjetničke tvorevine. Recimo, zamislite situaciju u kojoj korisnik velikog jezičnog modela zatraži od njega generiranje logotipa za njegovu privatnu tvrtku. Tko je onda vlasnik toga logotipa, to jest, čija su autorska prava tog logotipa? U situaciji bez korištenja velikih jezičnih modela taj posao odrađuju pojedinci kojima pripadaju autorska prava na njihov proizvod te su za to plaćeni. Ne samo da veliki jezični modeli u tom slučaju preuzimaju poslove tih pojedinaca već čine porijeklo samog proizvoda nejasnim i netipičnim. Moje mišljenje je da će se kroz naredne godine smanjiti potreba za redundantnim poslovima koje će zamijeniti umjetna inteligencija te da će se ljudi moći fokusirati na kreativnije zadatke. Što se tiče kreativnih radova, smatram kako umjetna inteligencija ne smije imati doticaja sa kreiranjem umjetničkog sadržaja. Umjetnički sadržaj mora biti djelo umjetnika koji je kroz svoj osobni dojam dočarao publici svoju viziju.

Duboke neuronske mreže

Duboke neuronske mreže su programi strojnog učenja koji oponašaju rad ljudskog mozga. U ljudskom mozgu neuroni su međusobno povezani tzv. Sinapsama [8]. Neuronske mreže oponašaju sinapse sa realnim brojevima koji predstavljaju težinu tojest jakost veze između dva neurona, slično kao i u ljudskom mozgu gdje su češće korišteni neuroni jače povezani. Unutar duboke neuronske mreže nalazi se velik broj neurona koji su poredani u stupce.

Jedan takav stupac ima n neurona od kojih je svaki povezan sa svakim neuronom iz slijedećeg i prošlog stupca kao što je prikazano shematskim prikazom na slici 3.



Slika 3. Shematski prikaz duboke neuronske mreže.[3]

Takve stupce možemo predstaviti težinskom matricom koja ima dimenzije NxM gdje N predstavlja veličinu prošlog stupca, a M veličinu slijedećeg stupca. Obično su N i M iste veličine. Kada treniramo tojest učimo neuronsku mrežu zapravo podešavamo težine između neurona. Na taj način naša neuronska mreža nakon treninga ima neurone čije su težine podešene tako da za dani ulaz u sustav daje optimalni put prema izlazu te na kraju dobivamo konačni vektor težina (engl. embedding) [3]. Između svakog stupca neurona nalazi se funkcija koja aktivira neurone i omogućuje prijenos signala između slojeva. Takve funkcije nazivamo aktivacijskim funkcijama.

Aktivacijske funkcije

Aktivacijske funkcije služe za unos nelinearnosti u sustav. Kada ih sustav ne bi koristio dolazilo bi do pojave simetrije između neurona u pojedinom sloju te bi za posljedicu imali više neurona koji daju isti izlaz za isti ulaz. Takve situacije pokušavamo izbjeći jer želimo dobiti neurone koji mogu:

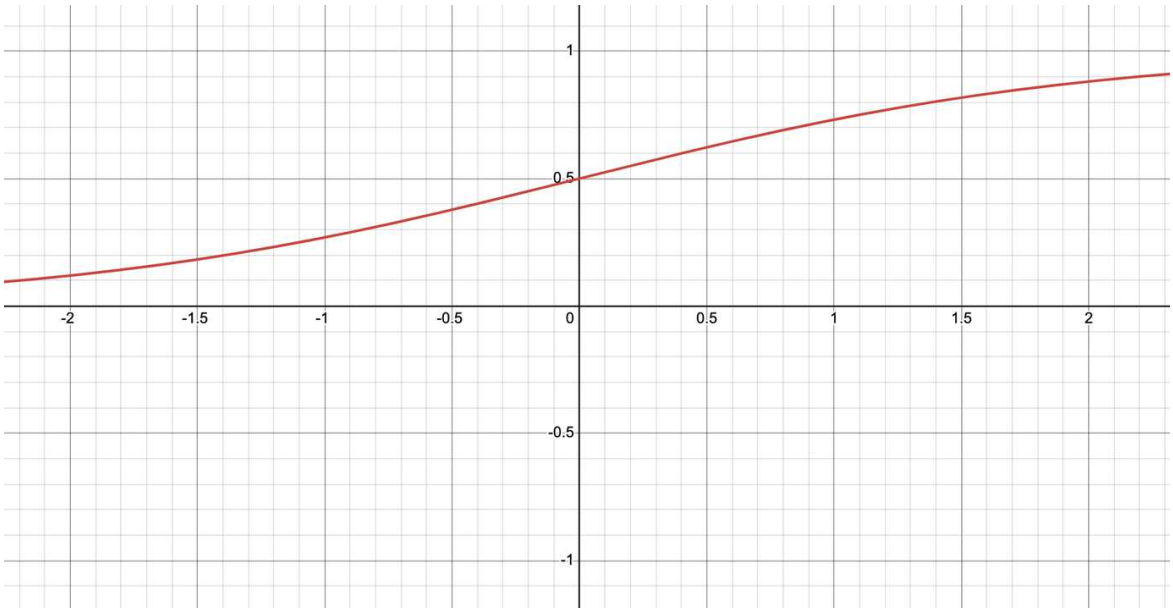
- Prepoznavati različite značajke
- Unositi raznolikost u mrežu
- Osigurati efikasnu obuku
- Podržavati nelinearnost

Postoje razne aktivacijske funkcije, a neke od najčešćih su slijedeće [9].

Sigmoidna funkcija

Sigmoidna funkcija kompresira ulazne vrijednosti u rasponu od 0 do 1. Često se koristi za binarne klasifikacijske zadatke. Grafički prikaz sigmoidne funkcije prikazan je na slici 4, a formula glasi:

$$\sigma(x) = \frac{1}{1+e^{-x}} \quad \{4\}$$

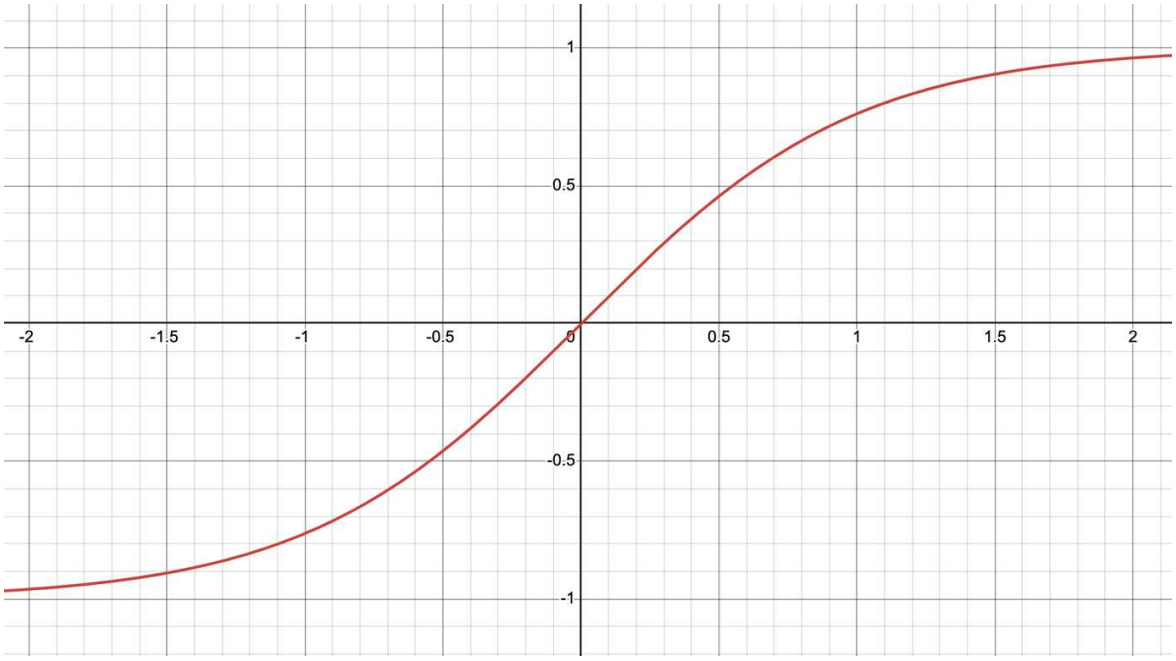


Slika 4. Grafički prikaz sigmoidne funkcije. Alat: Desmos

Tangens hiperbolički

Funkcija tangens hiperbolički kompresira ulazne vrijednosti u rasponu od -1 do 1 te ima bolje performanse jer je centralnosimetrična s obzirom na ishodište tojest centrirana u vrijednosti nula. Grafički prikaz funkcije tangens hiperbolički vidljiv je na slici 5, a formula glasi:

$$\tanh(x) = \frac{e^{x}-e^{-x}}{e^{x}+e^{-x}} \quad \{5\}$$



Slika 5. Grafički prikaz funkcije tangens hiperbolički. Alat: Desmos

Funkcija skoka

Funkcija skoka ili step funkcija daje izlaz 1 ili 0 i koristi se za binarnu klasifikaciju. Velik nedostatak ove funkcije je to što nije diferencijabilna u točki x = 0 te se određene metode treniranja ne mogu koristiti ukoliko se koristi ova aktivacijska funkcija. Međutim pogodna je u slučajevima kada nam je potrebna brzina. Formula funkcije skoka glasi:

$$f(x) = \begin{cases} 1 \; ako \; je \; x \; \geq 0 \\ 0 \; ako \; je \; x < 0 \end{cases} \quad \{6\}$$

Treniranje

Postoji više načina treniranja neuronske mreže. Treniranjem neuronske mreže želimo podesiti težine između neurona tako da daju željeni izlaz za određeni ulaz. Prije treninga obično postavljamo vrijednosti težina na nasumične brojeve kako bi izbjegli simetriju između neurona. Na taj način osiguravamo da naš izlaz bude nelinearan i efikasan.

Metoda mješavine stručnjaka (MoE)

Metoda mješavine stručnjaka (engl. Mixture of Experts, MoE) je metoda strojnog učenja. Model umjetne inteligencije je podijeljen na više podmreža, tzv. Stručnjaka [4]. Svaki stručnjak specijaliziran je za obradu određenog dijela podataka te na taj način daje bolji rezultat nego monolitni modeli umjetne inteligencije. Metoda mješavine stručnjaka uvelike smanjuje troškove računanja tijekom faze treniranja te vrijeme zaključivanja gotovog

modela. Trenutni aktualni model koji koristi metodu mješavine stručnjaka je Mixtral 8x7B koji koristi osam stručnjaka od kojih svaki ima sedam milijardi parametara.

Zašto MoE umjesto monolitnog modela?

Neuronske mreže teže prema sve većem broju neurona, parametara te je stoga potreba za računalnim resursima sve veća. U monolitnim modelima povećavanje parametara znači lošije vremenske performanse. Ukoliko uzmemo monolitni model sa 56 milijardi parametara, za svaki ulaz u sustav, algoritam će morati proći kroz svih 56 milijardi parametara da bi dao konačan izlaz. No, ukoliko uzmemo model koji koristi metodu mješavine stručnjaka sa istim brojem parametara, ali razdijelimo ga na osam stručnjaka, algoritam će morati proći kroz 1/8 (ukoliko se samo jedan stručnjak koristi za jedan ulaz) ukupnih parametara, odnosno sedam milijardi što je značajno brže i efikasnije. Svaki stručnjak u MoE specijaliziran je za svoju poddomenu ulaza te je s manje parametara jednako dobar ili i bolji nego monolitni model s mnogo više parametara zbog njegove opširnost.

Kako radi MoE?

Metoda mješavine stručnjaka mora odrediti koje će stručnjake koristiti. Za to služi razdjelnik (engl. router) [4]. Prilikom svakog ulaza u sustav algoritam pomoću razdjelnika bira koje će stručnjake koristiti. Tipično koristi više stručnjaka, na primjer dva, kako bi dobio točniji i bolji odgovor. Stručnjaci se biraju prema više kriterija i ovise o samom modelu, no neki osnovni bi bili specijalizacija stručnjaka (npr. stručnjak za tehnologiju), raspoloživost i dostupnost. Stručnjaci su zapravo zasebni modeli umjetne inteligencije trenirani da obavljaju određeni zadatak. Parametri koje koristi sustav s metodom mješavine stručnjaka mogu biti dijeljeni između različitih stručnjaka. Dijeljeni parametri su obično oni od kojih svi ili nekoliko stručnjaka imaju koristi. Takvi parametri mogu biti oni koji su zaduženi za samosvjesnost modela.

Mixtral 8x7B

Mixtral 8x7B trenutno je najpopularniji model umjetne inteligencije koji koristi metodu mješavine stručnjaka. Podržava pet jezika: Engleski, Njemački, Francuski, Španjolski i Talijanski [7]. Model se sastoji od osam stručnjaka koji koriste po sedam milijardi parametara. Međutim neki od tih parametara se ponavljaju, konkretno parametri za samosvjesnost. Stoga iako bi netko mogao zaključiti da ovaj model ima 56 milijardi

parametara ukupno, zapravo se radi o 46.7 milijardi. Performanse ovog modela nadmašuju Llama2 model kao i GPT-3.5 u gotovo svim mjerilima što je vidljivo u tablici performansi na slici 6.

	LLaMA 2 70B	GPT - 3.5	Mixtral 8x7B
MMLU (MCQ in 57 subjects)	69.9%	70.0%	**70.6%**
HellaSwag (10-shot)	87.1%	85.5%	86.7%
ARC Challenge (25-shot)	85.1%	85.2%	**85.8%**
WinoGrande (5-shot)	**83.2%**	81.6%	81.2%
MBPP (pass@1)	49.8%	52.2%	**60.7%**
GSM-8K (5-shot)	53.6%	57.1%	**58.4%**
MT Bench (for Instruct Models)	6.86	**8.32**	8.30

Slika 6. Tablični prikaz usporedbe vrijednosti performansi modela Mixtral 8x7B s LLaMA 2 70B i GPT-3.5. [7]

Prompting

„AI prompting refers to the process of interacting with an artificial intelligence (AI) system by providing specific instructions or queries to achieve a desired outcome.“ [10]. Prompting je tehnika pisanja najčešće tekstualnog ulaza za model umjetne inteligencije. Na ulaz modela umjetne inteligencije želimo dovesti instrukcije modelu te mu objasniti što želimo od njega. Najčešća uporaba promptinga je u svrhu obrade prirodnog jezika (engl. Natural Language Processing, NLP). Prompt mora biti pomno sastavljen kako bi model razumio što od njega tražimo. Želimo mu dovesti samo relevantne informacije i ne prenapućiti ga njima jer to može dovesti do kontra efekta i neželjenog odgovora. Prompting još možemo definirati kao postavljanje specifičnih upita i instrukcija modelu umjetne inteligencije kako bi ga usmjerili prema željenom zadatku ili kako da poslane podatke interpretira. Prompting je bitan dio ovog rada jer je neizbježan kod slanja upita na korištene modele te usmjeravanja modela ka željenom zadatku.

Zašto je prompting bitan?

Prompting ima široku primjenu u generiranju teksta i procesuiranju prirodnog jezika. Najčešće primjene promptinga su slijedeće:

Preciznost odgovora

Cilj je usmjeriti model umjetne inteligencije ka što točnijem odgovoru uz minimalni trošak. Kako bi izbjegli potrebu treniranja modela za specifičnu primjenu, usmjerit ćemo model ulaznim instrukcijama tojest ulaznim tekstom ka željenom rezultatu, a pritom koristiti model koji je sposoban generirati željeni izlaz. Koristit ćemo modele umjetne inteligencije koji imaju dovoljno informacija o problemu koji želimo riješiti.

Brzina

Napravimo li dovoljno dobar prompt, koji će naš model lako razumjeti, ubrzat ćemo proces dohvaćanja odgovora. Pri izradi prompta pokušavamo koristiti što jasnije i preciznije upute kako bismo osigurali da model odmah generira željeni rezultat. Ovo smanjuje potrebu za dodatnim iteracijama i popravcima, čime se povećava efikasnost.

Bolje korisničko iskustvo

Postavimo li neko generalno pitanje, model umjetne inteligencije bi mogao generirati hrpu teksta od kojeg nas efektivno zanima isključivo nekoliko linija. Ukoliko mu damo preciznije instrukcije odgovor će biti generiran sukladno instrukcijama te ćemo dobiti isključivo informacije koje nas zanimaju.

Fleksibilnost i prilagodljivost

Pomno osmišljeni promptovi omogućuju da koristimo generičke modele za neko usko područje primjene. Nema potrebe za prevelikim testiranjem dobivenih odgovora što omogućuje brzu prilagodbu modela na različite primjene.

Detekcija disruptivnih otkrića i inovacija

Detekcija disruptivnih otkrića i inovacija je proces pronalaženja novih tehnologija, inovacija ili poslovnih modela koji imaju potencijala da značajno promjene tržište ili industriju. Obično su to male tvrtke koje su tek u svojim početcima te su nam one i najzanimljivije zbog investicijskih prilika. Pronalaženje takvih otkrića zahtjeva praćenje tržišta i trendova, no često te male tvrtke prolaze nezamijećeno jer nemaju financijsku snagu za uzdrmavanje tržišta u kratkom vremenskom periodu. Stoga bi alat koji automatizirano može prolaziti kroz patente, znanstvene članke ili slično igrao veliku ulogu u pronalaženju takvih inovacija.

Problem pretraživanja takvih otkrića po vijestima je taj što trenutak kada je vijest objavljena, sam patent ili inovacija je već poznata. Automatiziranom metodom pokušavali bi pronaći takve patent prvom mogućom prilikom nakon objave rada o inovaciji.

Analiza startup-a

Još jedno od rješenja detekcije disruptivnih otkrića je praćenje i analiza startup-a. Startup-i su tvrtke koje su u svojim početcima te njih smatramo najzanimljivijima jer potencijalno ulaganje u njih nam može donijeti najveće dobitke. Pomno prateći njihov rad možemo zaključiti predstavlja li pojedina firma potencijal za ulaganje. Problem praćenja startup-a je taj što ih ima mnogo i bez adekvatnog automatiziranog alata teško je pronaći tvrtku vrijednu ulaganja.

Povratne informacije od korisnika

Ukoliko možemo zaključiti da su povratne informacije od korisnika relevantan izvor informacija, one nam mogu poslužiti kako bi se odlučili za ulaganje u pojedini proizvod. U mnogim slučajevima moguće je provoditi ankete i upitnike nad korisnicima. Takva analiza i anketiranje moraju biti pomno strukturirani kako bi od korisnika dobili što točnije informacije. Dobijemo li krive povratne informacije od korisnika, mogli bi pogrešno zaključiti stanje startup-a ili proizvoda i potencijalno donijeti krivu odluku o investiciji. Analiza društvenih mreža sve se češće koristi zbog svoje jednostavnosti i cijene. Velik broj korisnika određenog proizvoda služi se internetom te tim putem od njih možemo tražiti povratnu informaciju.

Primjena umjetne inteligencije u detekciji disruptivnih otkrića

Umjetna inteligencija sve se češće koristi za rješavanje mnogih problem u svakodnevnom životu te je primjenjiva i na ovo područje. Sustav koji bi mogao detektirati disruptivna otkrića i inovacije bio bi od velikog značaja za investitore. Sustavi umjetne inteligencije mogu obraditi ogromne količine podataka te je stoga primjena AI-a na ovakav problem izuzetno perspektivno rješenje automatizacije. Modeli umjetne inteligencije mogu biti trenirani tako da prepoznaju neobične riječi i fraze što bi moglo upućivati na nove tehnologije. Također, umjetna inteligencija može pomoći u klasifikaciji inovacija i disruptivnih otkrića te ih rangirati prema disruptivnom potencijalu. Postoje mnogi faktori po kojima bi se takva klasifikacija mogla provesti. Neki od faktora su: tržišni potencijal, tehnološka inovativnost i skalabilnost patenta.

Eksperiment

Skup podataka

Skup podataka (engl. dataset) nad kojim je proučavana metoda mješavine stručnjaka i usporedba s monolitnim modelom je pomno izrađen skupivši 271 720 članaka sa interneta. Te članke smo različitim metodama obrađivali te na kraju dobili 35 400 obrađenih članaka među kojima je 21.28% signala, odnosno 7534 signala.

Struktura skupa podataka

Skup podataka zapisan je u obliku JSON (JavaScript Object Notation) datoteke jer je takav oblik datoteke prikladan za pohranjivanje objekata kao zamjena za korištenje relacijske baze podataka. Uz to ovaj format je prikladan za razmjenu podataka s udaljenim poslužiteljima na koje smo se povezivali kako bi obradili podatke. Konačni skup podataka u JSON formatu zauzima 224,1 MB. Struktura JSON datoteke je slijedeća:

- ID — jedinstveni identifikator članka
- Naslov
- Datum objave
- Tekst
- Link — poveznica na mjesto gdje je članak objavljen
- GPT-3 signal — vrijednost 0 ili 1 dobivena prilikom obrade skupa podataka
- Teme
- Rast — rast u interesu pojedinih tema nakon objave članka
- Signal — konačna vrijednost 0 ili 1 koja indicira da li je članak signal

Prikupljanje podataka

Podaci su prikupljani s internetskih stranice koje objavljuju članke kao što su „BBC“, „The Guardian“, „TechCrunch“, „Wccftech“ i „Reuters“. Prikupljeno je 271 720 članaka koji su naknadno filtrirani.

Obrada podataka

Obrada podataka je najzahtjevniji dio ovog rada. S obzirom na količinu početnih podataka ovaj dio završnog rada je vremenski zahtjevan te potencijalno zahtjevan i u financijskom smislu. Problematika skupljanja podataka nad ovakvom domenom jest ta da inicijalno

skupljanje podataka preko internetskih stranica vrlo često ne garantira da ćemo dobiti skup podataka koji će sadržavati dovoljno signala kako bi adekvatno mogli koristiti metodu mješavine stručnjaka. Naime potrebno je nekom metodom filtrirati originalno skupljene podatke i sastaviti manji podskup podataka koji će dalje ići na obradu. S približno 300 tisuća članaka, povezivanje na API sučelje nekog modela umjetne inteligencije, konkretno OpenAI API, trajalo bi predugo i koštalo previše. Stoga smo morali koristiti neke alternativne metode za filtriranje podataka kako bi manji podskup mogli obrađivati. Odlučili smo se za klasifikaciju teksta pomoću FastText biblioteke razvijenu od strane „Facebook AI Research Lab“ [11]. Pomoću FastText-a za svaki članak definirali smo njegovu vektorsku reprezentaciju te smo na taj način mogli izabrati najbolje članke te odbaciti „smeće“. Slijedeći korak je prikupiti teme i bitne fraze iz teksta te pitati GPT-3 model predstavlja li članak signal. Taj korak odrađen je preko OpenAI API sučelja korištenjem GPT-3-Turbo model. Kako bi nam GPT-3 vratio relevantne podatke složen je tekstualni upit (engl. prompt):

> „You are an assistant for signal search and keyword extraction. Given the provided article (in JSON format with multiple fields), you need to determine if it is a signal. An article is considered a signal if it presents significant news for its time and has subsequently demonstrated lasting impact, successful implementation, or recognized advancement in the respective field. Signals are defined as unusual or exceptional instances that deviate from the norm or expected patterns. Your task is to analyze the provided text and respond with ONLY a "1" if it exhibits positive anomalies meeting the specified criteria, or ONLY a "0" if it does not. If you encounter the same or a very similar article again, you should give an IDENTICAL result as you gave when you first saw the article. The information should be groundbreaking for its time period and have shown sustained impact or advancement beyond the initial publication. Ensure that the news is of crucial matter and impactful to the entire sector/industry or the world. After determining if it is a signal or not, extract keywords (not exceeding 10) and general topics from the article text, such as "Technology", "Economic", etc., for topics, and keywords like "Bitcoin", "Alexa", "talking assistant", etc. The topics and keywords should be provided in an array of strings. Your output should consist of 2 lines: the first line with either 0 or 1 indicating if it is a signal, and the second line with an array of keywords and topics (in format ["topic1", "topics2", "keyword1", ...]).“

Ovakav upit na tekstualni model umjetne inteligencije omogućavao nam je da dobijemo kvalitetnu povratnu informaciju u formatu koji možemo „parsirati“ i pohraniti. Ovaj korak trajao je oko dva dana kako bi dobili sve tražene povratne informacije za filtrirani skup

podataka. Nakon toga, za svaku temu ili frazu bilo je potrebno pronaći rast interesa. Ovdje je postojalo nekoliko metoda s kojima smo mogli pristupiti rješavanju ovog problema. Jedna od njih je bila gledati financijsko stanje patenta koji se spominje u članku, no ovakav pristup ima svoje mane. Glavna mana je ta da nisu svi članci koje obrađujemo namijenjeni razvoju patenata, no ta vijest i dalje može biti signal. Uz to vrlo je zahtjevno prikupiti točne informacije o financijskom stanju patenata ili određenih tvrtki koje su ga razvile, a da to pritom bude automatizirano. Naše rješenje je to da koristimo Google Trends kao informaciju o interesu za određene teme i fraze. Stoga smo napravili program koji će čitati informacije o interesu za teme i fraze koje mu šaljemo. Zanima nas rast ili pad interesa za poslani podataka kroz prvih tri, pet i deset godina od objave članka, ako te informacije postoje. Naš skup podataka složen je od starijih članaka, 10 do 15 godina starih, kako bi mogli prikupiti interes kroz godine i na pametniji način zaključiti je li nešto što je inicijalno bio signal zaista signal.

GPT-3.5-Turbo

Model umjetne inteligencije je model koji je optimiziran za razumijevanje i generiranje prirodnog jezika i programskog koda te je prilagođen za razgovor s korisnikom. Veličina konteksta koji može primiti kao ulaz je 16 385 tokena [5].

Usporedba modela

Kao što je već navedeno usporedit ćemo rezultate koristeći dva modela umjetne inteligencije. Mixtral 8x7B koji koristi metodu mješavine stručnjaka i GPT-3.5-Turbo koji je monotoni model. Kako bi dobili rezultate moramo složiti python skriptu ili jupyter bilježnicu na Google Colabu kako bi se naš model pokretao na serverima od Google-a zbog boljih performansi i veće memorije. Jupyter bilježnica korištena je za Mixtral 8x7B, dok je za GPT-3.5-Turbo korištena lokalna python skripta jer ovdje nije problem memorija ukoliko se povezujemo na API sučelje od OpenAI-a. Koristimo isti prompt za oba modela kako bi točnije usporedili ponašanje modela. Prompt je slijedeći:

> „You are a signal detector. Your task is to determine if the given input, which is the text of an article, is a signal or not. A signal is something that is first seen, a groundbreaking achievement, or huge news about something that will potentially change some industries or politics. An example would be the early occurrence of an article about quantum computers that could work at temperatures close to 0 degrees Celsius, or that the USA goes to war with Russia. If something is a signal, respond with 1, and if it is not, respond

with 0. Do not respond with anything else. Use provided input to determine if it is a signal or it is not.“

Povezivanje na OpenAI API sučelje nije besplatno te ovisi o modelu koji se koristi. Navedene su cijene nekih modela u tablici 1. Metrika za naplatu poziva njihovog sučelja su tokeni i uglavnom se računaju u milionima tokena po jedinici cijene u dolarima $ [6].

Tablica 1 Prikaz cijena različitih GPT modela [6]

model	ulaz	izlaz
gpt-4o	$5.00 / 1M tokens	$15.00 / 1M tokens
gpt-4o-2024-05-13	$5.00 / 1M tokens	$15.00 / 1M tokens
gpt-3.5-turbo-0125	**$0.50 / 1M tokens**	**$1.50 / 1M tokens**
gpt-3.5-turbo-instruct	$1.50 / 1M tokens	$2.00 / 1M tokens

Pri računanju ukupne cijene potrebno je uzeti u obzir prompt koji smo složili. Ukoliko je prompt prevelik velik udio ukupne cijene otići će samo na prompt.

GPT-3.5-Turbo skripta

Pri izradi skripte za povezivanje na OpenAI sučelje moramo voditi računa o tome da pogreške u parsiranju odgovora moraju biti minimalne. Inače svaki takav poziv može rezultirati gubitkom rezultata i gubitkom financijskih sredstava koje smo uložili u taj poziv. Moramo biti svjesni da naš prompt koji smo napravili mora osigurati u većini slučajeva (minimalno 98%) da ćemo od našeg modela dobiti točni format kako bi ga mogli parsirati. Najveća pažnja mora biti posvećena minimiziranju mogućih pogrešaka i pokušaja oporavka od njih kako naši pozivi API sučelju ne bi bili beskorisni. Greške u skripti mogu biti minimalne, ali i dalje izazivati ogromne probleme. Jedan takav primjer tokom rada na ovom projektu bili su navodnici koje je generirao model, jednostruki umjesto dvostrukih. Kod ovolike količine podataka trebamo paziti na vrijeme koje je potrebno da se svi podatci evaluiraju. Približno 1 sekunda (točnije 0.89s) u prosjeku je potrebna za svaki odgovor. Ukoliko sve prođe dobro otprilike 9 sati je potrebno da prikupimo informacije o signalima za cijeli skup podataka. Još jedan detalj koji je vrijedno spomenuti je taj da se mora voditi računa o limitu koji imamo na OpenAI platformi s obzirom na financijska sredstva koja uložimo. Stoga je bitno napraviti detaljnu analizu skupa podataka i što točnije izračunati koliko nam je sredstva potrebno. Isto tako pametno je uložiti oko 10% više sredstva od planiranog kako bi se izbjegli kasniji problemi.

Konfiguracija

Potrebno je stvoriti virtualno okruženje za python skriptu. Virtualno okruženje nam nudi mogućnost da pokrenemo našu skriptu u bilo kojoj verziji pythona. Takvo okruženje i pokretanje na određenim verzijama pythona je esencijalno za ovaj završni rad jer nam mnoge biblioteke zahtijevaju određene verzije pythona. Nama je bila potrebna jedna biblioteka za spajanje na OpenAI API sučelje, no prilikom izrade skupa podataka bilo je potrebno instalirati i dodatne kao što je biblioteka za FastText. Python verzija 3.11.7. podržavala je sve biblioteke koje su bile potrebne.

Jupyter bilježnica za Mixtral 8x7B

Glavni razlog korištenja bilježnice za ovaj model je taj što model preuzimamo i pokrećemo sami, a za GPT-3.5-Turbo se spajamo na njihov model koji OpenAI pokreće. Velika količina računalnih resursa, pogotovo grafičkih kartica, potrebna je kako bi se jedan model ove veličine pokrenuo. Stoga je najbolja opcija pokrenuti takav model na udaljenim serverima, konkretno Google-ovim serverima. Prednost preuzimanja modela lokalno je ta što odgovore dobivamo znatno brže nego kada se spajamo na udaljeno sučelje nekog modela. Google Colab nam nudi razne resurse njihovih servera koje možemo koristiti pri pokretanju ovakvih modela, no za ovakav zadatak dovoljna je besplatna standardna verzija. Ukoliko bismo htjeli trenirati model ove veličine, trebali bismo plaćati dodatne računalne jedinice i memoriju. Ovdje imamo malo veću slobodu što se tiče pogrešaka jer koristimo besplatnu verziju Google Colaba te neće biti tolika šteta ukoliko se dogodi pogreška.

Prednosti korištenja Jupyter bilježnice

Neke prednosti kao što su korištenje Google-ovih resursa su već navedene, no postoje i mnoge druge. Jupyter bilježnica omogućuje programiranje u blokovima. To znači da se jedan programski blok može koristiti neovisno o drugome. Ovo je posebno korisno kada se radi sa modelima koje moramo učitati i instalirati unutar našeg okruženja. Ukoliko moramo izmijeniti dio koda ne moramo ponovno pokretati sve ostalo, posebno učitavanje modela, već izmjenjujemo samo onaj dio koji zahtjeva izmjenu. To uvelike olakšava i ubrzava rad nad modelima umjetne inteligencije. Dodatna prednost je dokumentiranje koda. U klasičnim programima pišemo komentare koji su teško čitljivi. Jupyter nudi opciju pisanja takvih komentara u formatiranom obliku tako da svaki dio koda možemo pojasniti sebi, ali i drugima, u stiliziranom obliku koji je lakši za čitanje.

Konfiguracija

Konfiguriranje okruženja na Google Colabu koristeći Jupyter bilježnicu je vrlo jednostavno. Sama bilježnica dolazi sa već konfiguriranim bibliotekama koje se najčešće koriste za rad s modelima umjetne inteligencije. Za ovaj rad dodatno je korišten T4 GPU koji nam Google Colab besplatno nudi. To znači da će se naš model pokretati na grafičkim karticama koje nam Google dodijeli, te posljedično ubrzati naš program. Uz to potrebna je instalacija nekih osnovnih biblioteka koje nam HuggingFace nudi za rad sa raznim modelima. HuggingFace je platforma preko se objavljuju gotovo svi „open source“ modeli umjetne inteligencije i besplatni su za korištenje uz račun na njihovoj platformi.

Rezultati

Dobiveni rezultati primjenom metode mješavine stručnjaka i primjenom monolitnog modela umjetne inteligencije su veoma slični. Iako je očekivanje bilo da će metoda mješavine stručnjaka dati znatno bolje rezultate, rezultati su se pokazali podjednakim. Razlog tome je uglavnom domena problema koji rješavamo. Disruptivna otkrića i inovacije su vrlo uzak pojam za modele umjetne inteligencije koji su prvenstveno trenirani kako bi rukovali s generiranjem teksta. Ukoliko bi htjeli poboljšati točnost i jednog i drugog modela, trebali bi ih dotrenirati za rad sa specifičnim podacima koje smo skupili. Glavni problem je taj da je teško odrediti je li nešto signal, a da pritom nemamo dodatne informacije o proizvodu van samo teksta koji naši modeli provjeravaju. Preciznost odgovora prikazana je u tablici 2.

Tablica 2 Preciznost određivanja signala i ne-signala dva istraživana modela

Mixtral 8x7B	GPT-3.5-Turbo
79.12%	79.04%

Dakle, rezultati su gotovo identični. Naš skup podataka sastoji se od približno 20% signala te da bi dobili bolje rezultate potrebno je naš skup podataka popuniti sa dodatnim signalima. Trenutno je oko 80% ne-signala te je vrlo jednostavno odrediti da li je nešto ne-signal. Puno teže je razlučiti je li tekst signal. U ovom djelu su testirani modeli bili značajno lošiji nego gledajući ukupan skup podataka. Naime više od pola primjera signala naši modeli su krivo protumačili kao ne-signalne. Preciznost za signale prikazana je u tablici 3.

Tablica 3 Preciznost određivanja signala dva istraživana modela

Mixtral 8x7B	GPT-3.5-Turbo
42.01%	48.51%

Ovdje možemo vidjeti kako je GPT-3.5-Turbo znatno bolje odredio signale, no ni jedan model nije dao dovoljno dobre i dovoljno precizne rezultate. Ukoliko bi ukupna točnost za određivanje signala bila oko 80% rezultati bi bili zadovoljavajući.

Evaluacija performansi

Oba modela loše prepoznaju signale na temelju zadanih instrukcija. Najbolji način za poboljšanje performansi je dotreniravanje modela za specifični zadatak. Na taj način umjesto modela umjetne inteligencije koji je sposoban rješavati razne zadatke, dobit ćemo model koji je specijaliziran za obradu specifičnih zadataka te bi to moglo biti rješenje i put

ka boljim performansama. Naš skup podataka svakako treba popuniti s još signala kako bi mogli dobiti precizniji omjer točnosti rezultata. Budući da koristimo binarnu klasifikaciju rezultata (dakle ili je signal ili nije) te imamo nešto manje od 80% ne-signala, da je naš model svaki puta odgovorio sa 0, odnosno da nije signal, dobili bi gotovi isti omjer kao i sada.

Analiza korištenih podataka

Već je spomenuto kako naš skup podataka izgleda i od kuda su podaci prikupljani. Distruptivna otkrića i inovacije su sigurno nešto što takvi izvori nude. Ipak problem leži u tome da iako tražimo disruptivna otkrića ili inovacije mi moramo selektirati one koji su signali. Naši signali unutar skupa podataka su klasificirani od strane modela umjetne inteligencije, no kako bi dobili bolji skup podataka potrebno je prikupiti podatke sa provjerenih izvora koji se bave isključivo pretragom signala. Kada bi imali pristup dovoljnom broju takvih podataka mogli bismo napraviti iznimno dobar skup podataka za trening i testiranje budućih modela.

Zaključak

U ovom radu istražena je metoda mješavina stručnjaka koristeći model Mixtral 8x7B i uspoređena je s monolitnim modelom GPT-3.5-Turbo. Analiziran je velik skup podataka od 271 720 tekstualnih članaka od kojih je filtrirano i obrađeno 35 400. Rezultati pokazuju da je ukupna preciznost oba modela slična, Mixtral 8x7B ima točnost od 79.12%, a GPT-3.5-Turbo 79.04%. Iako je bilo očekivano da metoda mješavine stručnjaka daje bolje rezultate, pokazalo se da su rezultati gotovo identični zbog specifičnosti domene problema. Poseban izazov je bio prepoznavanje signala, gdje su oba modela pokazala loše rezultate. Preciznost određivanja signala bila je 42.01% za Mixtral 8x7B i 48.51% za GPT-3.5-Turbo, što ukazuje na potrebu za daljnjim poboljšanjima. Korištenje metode mješavine stručnjaka nije znatno poboljšalo rezultate i lako bi se moglo zaključiti da je Mixtral 8x7B imao lošije performanse. No, Mixtral 8x7B ima nekoliko puta manje parametara nego GPT-3.5-Turbo koji ima 175 milijardi. Uzevši to u obzir može se zaključiti kako je metoda mješavine stručnjaka za veličinu modela dala poprilično dobre rezultate. Dakle korištenje metode mješavine stručnjaka značajno može smanjiti broj parametara koje koristimo, a u isto vrijeme davati slične ili čak bolje rezultate. Skup prikupljenih podataka trebao bi imati više signala kako bi testiranje moglo biti još preciznije. Uz to skup podataka u idealnom slučaju trebalo bi prikupljati s izvora koji nude podatke o točno našoj domeni disruptivnih otkrića i inovacija kako bi sigurno dobili kvalitetne podatke. Ukoliko bi se koristio model Mixtral 8x22B od Mistral AI-a koji ima znatno više parametara nego Mixtral 8x7B moguće je da bi dobili točnije odgovore, no takav model zahtjeva puno više računalnih resursa i nije prikladan za ovaj rad s obzirom na tehnološka ograničenja. Korištenje ovih tehnologija na uskoj domeni problema zahtjeva dotreniranje modela kako bi dobivali točnije odgovore pogotovo za domenu kao što su distruptivna otkrića i inovacije. Glavni problem je u identifikaciji signala, posebno s obzirom na ograničen broj signala u skupu podataka. Uspješno prepoznavanje signala zahtijeva detaljniju analizu i često dodatne informacije koje nisu sadržane u tekstu. Dodatno u radu se može vidjeti primjena raznih tehnika obrade podataka od kojih je sigurno najuspješnija bila korištenje dodatnih velikih jezičnih modela u svrhu klasifikacije članaka. Korištenje velikih jezičnih modela za obradu podataka je svakako dobar potez, no treba imati na umu da ova metoda nije previše efikasna ukoliko imamo skup podataka koji je prevelik ili je pun loših podataka. Ukoliko je skup podataka prevelik, kao što je naš originalni skup od približno 270 000 podataka bio, cijena obrade svih podataka uveliko raste te premašuje trenutne mogućnosti. Stoga se za filtriranje i obradu podataka koristila

alternativa FastText kojom smo prikupili vektorske zapise svih članaka i uspjeli ukratko filtrirati najbolje članke. FastText je bio odličan alat za filtriranje nekolicine članaka koje smo smatrali smećem i koji nikako nisu trebali ući u konačni skup podataka. No s druge strane potencijalno smo izbacili i neke članke koji bi možda bili i bolji odabir za originalni skup podataka. Možemo zaključiti da je kombinacija svih navedenih metoda svakako dobar način obrade podataka i analize određenih modela umjetne inteligencije, no uz dovoljne resurse rezultati bi sigurno bili i bolji. Metoda mješavine stručnjaka će sigurno kroz naredne godine postati jedna od vodećih metoda za optimizaciju performansi umjetne inteligencije u različitim domenama.

Literatura

[1] IBM, What are large language models (LLMs)?, URL: https://www.ibm.com/topics/large-language-models, 10.06.2024.

[2] Sabrina Aquino, What are Vector Embeddings?, 06.02.2024., URL: https://qdrant.tech/articles/what-are-embeddings/, 10.06.2024.

[3] IBM, What is a neural network? URL: https://www.ibm.com/topics/neural-networks, 10.06.2024.

[4] IBM, What is mixture of experts? URL: https://www.ibm.com/topics/mixture-of-experts, 11.06.2024.

[5] OpenAI, Models, URL: https://platform.openai.com/docs/models/, 11.06.2024.

[6] OpenAI, Pricing, URL: https://openai.com/api/pricing/, 11.06.2024.

[7] Mistral AI Team, Mixture of experts, 11.12.2024. URL: https://mistral.ai/news/mixtral-of-experts/, 11.06.2024.

[8] Jan Šnajder, Marko Čupić, Bojana Dalbelo Bašić, Umjetne neuronske mreže, Zavod za elektorniku, mikroelektroniku i inteligentne sustave, Fakultet elektrotehnike i računarstva, 2008.

[9] Encord blog, Activation Functions in Neural Networks: With 15 examples, 25.07.2023. URL: https://encord.com/blog/activation-functions-neural-networks/#:~:text=Activation%20functions%20play%20a%20crucial,and%20numerous%20non%2Dlinear%20variants., 12.05.2024.

[10] What is AI prompting? URL: https://servicecenter.fsu.edu/s/article/What-is-AI-prompting#:~:text=AI%20prompting%20refers%20to%20the,to%20achieve%20a%20desired%20outcome., 12.06.2024.

[11] Facebook AI Research lab, FastText, URL: https://fasttext.cc/

Primjena metode mješavine stručnjaka u detekciji disruptivnih otkrića i inovacija

Sažetak

Metoda mješavine stručnjaka polako zamjenjuje kasične monolitne modele umjetne inteligencije zbog svojih performansa i preciznosti odgovora. Takvim pristupom svaki stručnjak u sustavu specializiran je za jedan podskup ulaznih podataka. Primjenom te metode znaćajno se smanjuje računalni trošak uz potencionalno bolje performanse. U ovom radu istražuje se primjena takve metode nad pronalaženjem disruptivnih otkrića i inovacija. Ovakav pristup je pogodan za probleme koji zahtjevaju maksimalnu točnost i preciznost. Usporedit će se rezultati dobiveni metodom mješavine stručnjaka i monolitnog modela umjetne inteligencije nad pripremljenim skupom podataka.

Ključne riječi: Metoda mješavine stručnjaka, Disruptivna otkrića, Inovacije, Umjetna inteligencija, Strojno učenje, Neuronske mreže

Application of mixture of experts method in detecting disruptive discoveries and innovations

Abstract

The mixture of experts method is slowly replacing classical monolithic artificial intelligence models due to its performance and precise outputs. With this approach, every expert in the system specializes in one subset of input data. Applying this method greatly reduces computational costs and potentially provides better responses. This research is based on applying this method for finding disruptive discoveries and innovations. This approach is suitable for problems that require maximal precision and accuracy. Results from the mixture of experts will be compared with results from classical monolithic artificial intelligence models using a prepared dataset.

Keywords: Mixture of Experts; Disruptive discoveries; Innovations; Artificial Intelligence; Machine learning; Neural Networks