

KUNGL. VITTERHETS
HISTORIE OCH
ANTI KVITETS AKADEMIEN

ÅRSBOK 2023

STOCKHOLM 2023

THE ROYAL SWEDISH ACADEMY OF LETTERS,
HISTORY AND ANTIQUITIES

ACADÉMIE ROYALE SUÉDOISE DES
BELLES-LETTRES, DE L'HISTOIRE
ET DES ANTIQUITÉS

KGL. SCHWEDISCHE AKADEMIE FÜR LITERATUR,
GESCHICHTE UND ALTERTÜMER



© De enskilda författarna och KVHAA, Stockholm 2023

Grafisk formgivning: Lars Paulsrud

witterhetsakademien.se

ISBN 978-91-88763-44-0, ISSN 0083-6796

Italgraf Media, 2023

INNEHÅLL

- 7 Preses hälsningstal
- 11 År 2023 tilldelade medaljer och priser

MINNESORD

- 19 Johan Söderberg
- 27 Sture Allén

FÖREDRAG

- 45 Anders Wästfelt: *Gränsen mot framtiden. Insikter om jorden, tiden och samhället*
- 65 Gunnar Almevik: *Kulturarv i Antarktis*
- 87 Marie-Louise Bech Nosch: *Odysseus garderob. En homerisk hjälte och hans kläder*
- 103 Joakim Nivre: *Naturliga språk och artificiell intelligens*
- 115 Jonas Granfeldt: *Utbildning i främmande språk då, nu och sen. Moroten, piskan eller predikan?*
- 133 Hanne Fjelde: *Demokrati i vågskålen*
- 141 Arne Jarrick: *Kunskapen har en början men inget slut – varför är detta inte självklart?*
- 155 Hanna Bäck: *Politisk polarisering som utmaning för demokratin*

- 169 Anders Jarlert & Lars Berglund: *Kyrkomusikens historia*
– finns den?
- 187 Bo Andersson: *En gåva till en drottning – Aurora*
von Königsmarcks Nordischer Weÿrauch

ÅRSBERÄTTELSE, FÖRTECKNINGAR ÖVER LEDAMÖTER,
AKADEMIFORSKARE OCH PERSONAL

- 207 Ständige sekreterarens årsberättelse
- 213 Vitterhetsakademiens kulturfastigheter
- 227 Årsberättelser från vissa forskningsföretag m.m.
- 239 Publikationer utgivna 20.3.2022–19.3.2023
- 241 Alfabetisk förteckning över Akademiens ledamöter
- 258 Akademiens ledamöter i invalordning
- 261 Ledamöternas fördelning på klasser
- 263 Avlidna ledamöter 20.3.2022–19.3.2023
- 264 Akademiens utskott och nämnder
- 270 Vitterhetsakademiens akademiforskare
- 271 Akademiens personal

JOAKIM NIVRE

Naturliga språk och artificiell intelligens

NATURLIGA SPRÅK ÄR de språk vi människor förvärvar under vår uppväxt och använder för kommunikation under hela livet, exempelvis svenska, finska och japanska. Artificiell intelligens, vanligtvis förkortat AI, studerar förmågan hos datorprogram och robotar att efterlikna människans intelligens, vari ingår förmågan att använda naturliga språk. En slående illustration av språkets centrala roll i mänsklig intelligens hittar vi redan hos en av AI-forskningens pionjärer, den brittiske matematikern Alan Turing. I den klassiska artikeln ”Computing machinery and intelligence” från 1950 beskriver Turing ett test som kan användas för att avgöra om datorer kan tillskrivas intelligent beteende och som i korthet går ut på att låta en människa kommunicera med två agenter, varav den ena är en maskin och den andra är en annan människa.

När vi inte längre med mer än slumpmässig precision kan avgöra vilken av de två agenterna som är mänsklig, måste vi enligt Turing tillskriva maskinen intelligent beteende, något som Turing själv trodde skulle inträffa före millennieskiftet. Man kan diskutera om detta är ett bra test på artificiell intelligens, men för mig som språkforskare är det intressanta att språk och språklig kommunikation tillskrivs en så central roll i resonnementet. Det är genom att tala med människor på människors språk som en maskin kan visa att den är intelligent, och det är genom att ställa

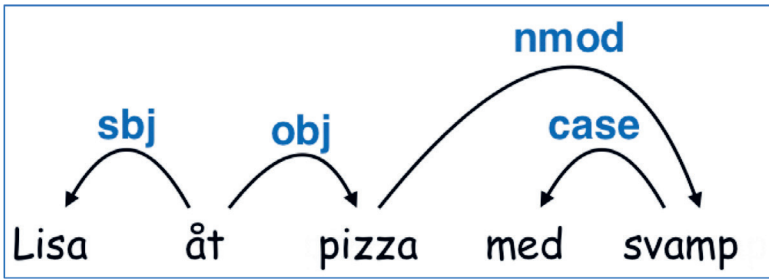
frågor på naturligt språk som vi människor kan testa maskinens kunskap och förmåga att dra slutsatser.

AI-relaterad forskning om språk bedrivs bland annat inom ämnet datorlingvistik, i dag ofta kallat språkteknologi eller NLP (en förkortning av engelskans ”natural language processing”), vilket varit mitt eget huvudsakliga forskningsområde under snart tre decennier. Datorlingvistik kan beskrivas som ett januslikt ämne, vars ena ansikte är att utveckla praktiska tillämpningar som översättningsprogram och digitala assistenter – Google Translate, Siri och ChatGPT är aktuella exempel – och vars andra ansikte är att vinna kunskap om naturliga språk genom att experimentellt pröva teorier och modeller om den mänskliga språkförmågan.

Hur dessa två sidor av ämnet hänger ihop är en komplex fråga som har getts olika svar vid olika tidpunkter. Är det så att lingvistiska teorier kan hjälpa oss att utveckla bättre praktiska system för exempelvis översättning? Är det tvärtom så att vi genom att studera vad som fungerar i praktiska tillämpningar kan utveckla bättre lingvistiska teorier? Eller är det rentav så att det inte finns någon direkt koppling mellan teoretisk kunskap om språket och praktiska tillämpningar av språkteknologi och AI? För tio år sedan skulle nog majoriteten av forskare ha sagt att AI-system för att kunna tolka och generera naturligt språk behöver innehålla moduler som svarar mot traditionella delområden av lingvistik, exempelvis en modul för morfologi som kan analysera ords struktur och en modul för syntax som kan analysera den grammatiska strukturen i meningar. I dag är detta inte alls lika självklart – vilket jag ska återkomma till i avslutningen – men innan dess vill jag göra två nedslag i min egen datorlingvistiska forskning som i stor utsträckning har handlat om syntax.

Det ena huvudspåret i min datorlingvistiska forskning har handlat om effektiva algoritmer för automatisk syntaktisk analys. Alltsedan Noam Chomskys banbrytande arbeten på 1950-talet har syntaxen setts som den centrala komponenten i språkets produktivitet, alltså vår förmåga att konstruera ett potentiellt oändligt antal meningar och att förstå meningar som vi aldrig tidigare har hört genom att tillämpa generella, rekursiva

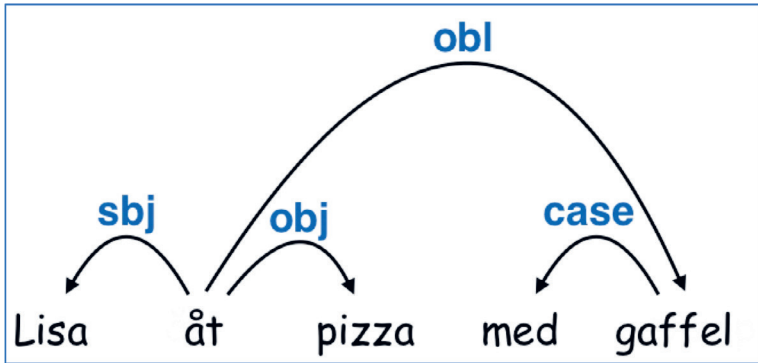
regler för hur ord kombineras till större enheter som fraser och satser. Det verkar därför rimligt att anta att en artificiell agent för att kunna använda naturliga språk på mänsklig nivå också måste kunna analysera språkets syntaktiska struktur. Därför har jag och andra forskare ägnat betydande tid åt att – för att uttrycka det enkelt – lära datorer att ta ut satsdelar. Givet en mening som *Lisa åt pizza med svamp* vill vi att datorn automatiskt ska kunna konstruera en representation av meningens syntaktiska struktur, vilket illustreras i figur 1 med ett så kallat dependensträd.



Figur 1. Dependensträd för *Lisa åt pizza med svamp* (sbj = subjekt, obj = objekt, nmod = nominal bestämning, case = kasusmarkör).

Vari består då problemet? Den största utmaningen ligger i det naturliga språkets flertydighet, i det här fallet strukturell flertydighet, vilket innebär att en sekvens av ord inte entydigt bestämmer den syntaktiska strukturen ens om vi känner till alla de ingående ordens kategorier. Om vi exempelvis byter ut ordet *svamp* mot ordet *gaffel*, förändrar detta vår tolkning av satsens struktur, så att den mest sannolika analysen nu är att det avslutande prepositionsuttrycket är bestämning till verbet *åt* snarare än till substantivet *pizza*. Det är ett adverbial som anger med vilket instrument Lisa äter, snarare än ett attribut som anger vilken sorts pizza det är frågan om, vilket framgår av dependensträdet i figur 2.

Flertydigheten är ett problem även ur ett datavetenskapligt perspektiv, eftersom den ökar den matematiska komplexiteten i problemet. Anta-



Figur 2. Dependensträd för *Lisa åt pizza med gaffel* (sbj = subjekt, obj = objekt, obl = oblik bestämning, case = kasusmarkör).

let möjliga strukturer växer exponentiellt med meningens längd, vilket innebär att en enkel algoritm som i tur och ordning söker igenom alla möjliga strukturer skulle bli alltför långsam för praktiska tillämpningar. Till detta kommer krav på robusthet, det vill säga att systemet måste kunna tolerera oväntade fenomen i de meningar som ska analyseras, såsom stavfel, utelämnade ord eller neologismer.

När jag började forska om automatisk syntaxanalys för tjugo år sedan byggde de flesta metoder på att man härleder alla analyser som är förenliga med en formell grammatik och rangordnar dessa med hjälp av en sannolikhetsmodell. Detta var för det första ineffektivt. Att analysera en mening kunde ibland ta flera sekunder, vilket omöjliggjorde storskaliga tillämpningar. Det var inte heller särskilt robust, eftersom formella grammatiker har låg tolerans för oväntad variation i indata. Dessutom var det inte förenligt med psykolingvistiska studier av mänsklig språkförståelse, vilka har visat att vi normalt går direkt på en bestämd analys och endast blir medvetna om flertydigheter när något går fel. Inspirerad av dessa resultat formulerade jag ett tämligen ambitiöst forskningsprogram med fyra krav som en algoritm för syntaktisk analys ska uppfylla:

- Robusthet: Varje mening ska erhålla *minst* en analys.
- Entydighet: Varje mening ska erhålla *högst* en analys.
- Korrekthet: Varje mening ska erhålla en *korrekt* analys.
- Effektivitet: Varje mening ska analyseras i *linjär* tid.¹

Det är värt att notera att en algoritm som samtidigt uppfyller kraven på robusthet och entydighet tilldelar varje mening *exakt* en analys. Om denna enda analys också är korrekt – i betydelsen att den överensstämmer med en modersmålstalares tolkning – och kan härledas i linjär tid, har vi nått vårt mål.

Den metod som jag och mina medarbetare utvecklade inom detta forskningsprogram kom att kallas transitionsbaserad dependensanalys, vilket kan låta komplicerat men vilket väsentligen bygger på två ganska enkla idéer: (i) att använda en så kallad girig algoritm för att uppnå robusthet, entydighet och effektivitet; och (ii) att förlita sig på maskininlärning från annoterade språkdata, snarare än formella grammatiker, för att uppnå så hög korrekthet som möjligt. Genom att kombinera dessa två idéer kunde vi skapa ett system som med avseende på korrekthet i analysen var i nivå med de bästa system som fanns sedan tidigare, men som var mer robust och framför allt tusentals gånger snabbare. Detta öppnade i sin tur dörren för att använda syntaktisk analys i storskaliga praktiska tillämpningar. Exempelvis introducerade Google transitionsbaserad dependensanalys både i sitt översättningssystem och i sin sökmotor för ungefär tio år sedan.

Hur fungerar då metoden? Om vi börjar med den giriga algoritmen, så bygger den på ett så kallat transitionssystem där vi har en buffert som vi fyller med meningens ord och en stack där vi kan mellanlagra orden under analysens gång. I varje ögonblick kan datorn göra en av tre saker: (i) flytta nästa ord till stacken; (ii) kombinera de översta orden på stacken med en relation åt höger; eller (iii) kombinera de översta orden på stacken med en relation åt vänster. Vi säger att systemet i varje ögonblick har tre (och endast tre) möjliga transitioner. Genom att mekaniskt utföra en sek-

vens av transitioner kan datorn processa alla orden i en mening och samtidigt bygga ett dependensträd som representerar meningens syntaktiska struktur. Till exempel kan dependensträdet i figur 1 konstrueras med en sekvens av transitioner som inleds med att systemet (i) flyttar ordet *Lisa* till stacken, (ii) flyttar ordet *åt* till stacken, (iii) kombinerar de två översta orden på stacken (alltså *Lisa* och *åt*) med en (subjekts)relation åt vänster.

Vad har vi då uppnått med att konstruera ett transitionssystem av detta slag? Man kan formellt bevisa (i) att varje sekvens av transitioner skapar ett dependensträd; (ii) att varje dependensträd² skapas av någon sekvens av transitioner; och (iii) att antalet transitioner är en linjär funktion av antalet ord. Härav följer att vi kan uppfylla alla våra fyra krav om vi bara korrekt kan förutsäga nästa transition i varje sekvens. Vi har därmed reducerat det komplexa problemet att analysera en hel mening till ett enklare problem – att förutsäga nästa transition givet det tillstånd som skapats av de tidigare transitionerna. Detta problem kan vi tackla med hjälp av maskininlärning, där vi försöker träna en statistisk modell för att förutsäga nästa transition givet det nuvarande tillståndet för stacken, bufferten och den partiella analys vi hittills har härlett. För att skapa en sådan modell behöver vi en träningsmängd av korrekta transitioner, vilket vi kan få från en stor korpus där alla meningar har annoterats med sin korrekta syntaktiska analys – en så kallad trädbank.

Transitionsbaserad dependensanalys är följaktligen ett exempel på en induktiv metod, där datorn lär sig i princip all kunskap om språket från exempel – snarare än formella regler. Ett viktigt villkor är dock att träningsdata, i det här fallet texter på ett naturligt språk, har försetts med lingvistisk annotering. Det är därmed också ett exempel på det som i maskininlärningssammanhang kallas övervakad inlärning, eller *supervised learning* på engelska. Detta är samtidigt en begränsning med metoden, eftersom lingvistisk annotering kräver mänsklig expertis – att en analys är korrekt betyder ju helt enkelt att den överensstämmer med den naturliga tolkningen för en människa – vilket gör det kostsamt att skapa riktigt stora datamängder. Detta för oss naturligt över till det andra huvudspåret

i min forskning som har handlat om att underlätta utvecklingen av anoterade korpusar – eller trädbanker – för världens språk.

För tjugo år sedan, när vi började experimentera med automatisk syntaktisk analys, var tillgången på syntaktiskt anoterade data tämligen begränsad, vad gäller både antalet språk och mängden data per språk. Dessutom saknades en enhetlig standard för annoteringen, vilket fick till följd att i stort sett varje språk hade sitt eget annoteringssystem baserat på en inhemsk grammatisk tradition. Avsaknaden av en standard försvårade forskningen på flera sätt, men framför allt var det mycket svårt att jämföra empiriska resultat från olika språk, eftersom man inte kunde veta om skillnader i kvalitet berodde på typologiska skillnader mellan språken eller på mer eller mindre godtyckliga skillnader i annoteringssystemen.

För snart tio år sedan var jag därför med och startade ett projekt med det uttryckliga målet att skapa en enhetlig standard för morfologisk och syntaktisk annotering och att skapa trädbanker för så många språk som möjligt. Inom detta projekt – som fick det något pretentiösa namnet Universal Dependencies – har vi utvecklat ett annoteringssystem bestående av (i) en uppsättning universella ordklasser; (ii) ett system av morfologiska särdrag som kan anpassas till olika språk; och (iii) en uppsättning universella syntaktiska relationer, med möjlighet att urskilja språkspecifika undertyper för att bättre kunna hantera den variation som finns i världens språk.

Projektet startade 2014 och den första datadistributionen inkluderade tio trädbanker från lika många språk. Vår avsikt var redan från början att göra detta till ett öppet samarbetsprojekt där alla som vill kan vara med och bidra i mån av resurser. Gensvaret från forskarsamhället har varit helt enastående – vilket tyder på att det faktiskt fanns ett behov av en universell standard. Den senaste datadistributionen, i november 2022, omfattade över 200 trädbanker med data från 138 olika språk, skapade av över 500 forskare från alla världens hörn, och tillsammans bestående av mer än 30 miljoner ord. Universal Dependencies, eller UD, var från början ett rent datorlingvistiskt projekt, med det primära målet att skapa

annoterade data som kan användas för maskininlärning och utvärdering av AI-system.

Med tiden har UD emellertid rönt allt större intresse också från lingvister, särskilt inom språktypologi och evolutionär lingvistik, som med hjälp av UD kan få ett mera finkornigt underlag för empiriska jämförelser mellan språk än det som erbjuds av traditionella typologiska databaser. Parallellt med detta har emellertid AI-forskningen och den mera tillämpade språkteknologin – som jag antydde i inledningen – utvecklats på ett sätt som gör att det i dag inte alls är självklart att vi faktiskt behöver lingvistiskt annoterade data för att lära datorer att använda naturliga språk. Innan jag avslutar vill jag säga några ord om den utvecklingen.

Under det senaste decenniet, och framför allt de senaste fem åren, har stora framsteg gjorts inom AI med hjälp av storskaliga neuronnät – statistiska modeller som i vissa avseenden är inspirerade av teorier om den mänskliga hjärnan – i kombination med superdatorer och enorma mängder data. Denna utveckling har även revolutionerat språkteknologin där vi sett en exponentiell ökning av modellernas storlek i takt med att it-företag som Google, Microsoft och Facebook har försökt överträffa varandra och tränat modeller som innehåller hundratals miljarder statistiska parametrar. En av de mest kända av dessa språkmodeller är GPT-3, skapad av OpenAI, vars språkförmåga har väckt såväl förundran som oro. Redan 2020 publicerade *The Guardian* en artikel med rubriken ”A robot wrote this entire article. Are you scared yet, human?”, som i stora stycken faktiskt var skriven av GPT-3 själv. GPT-3 är också grunden för dialogsystemet ChatGPT, som lanserades i november 2022 och som har fått enormt stor publicitet på kort tid.

Det är framför allt två saker som utmärker storskaliga språkmodeller som GPT-3. Den första är att de kan generera text av en sådan kvalitet att det ofta är omöjligt att avgöra om texten är producerad av en människa eller en maskin – därigenom kan de möjligen sägas klara en mycket begränsad variant av Turing's test. Den andra är att de kan tränas till denna förmåga utan att matas med någon form av explicit lingvistisk kunskap,

till exempel i form av anoterade data. För att förstå hur detta är möjligt behöver vi ännu en gång återvända till 1950-talet och en annan av pionjärerna inom datavetenskap och AI – Claude Shannon.

Shannon grundade informationsteorin, som bland annat handlar om hur mycket information som kan kodas i olika typer av system och hur effektivt denna information kan överföras i olika kanaler, och där begreppet entropi används för att mäta mängden information. Shannon var också intresserad av naturliga språk och konstruerade ett experiment för att mäta entropin i skriven engelska, vilket beskrivs i artikeln ”Prediction and entropy of printed English” från 1951. Experimentet går ut på att en person väljer ut en text och att de andra deltagarna ska försöka gissa texten bokstav för bokstav. Genom att räkna hur många gissningar som i genomsnitt behövs för att rekonstruera en text, kan man med hjälp av detta experiment uppskatta entropin i skriven text. Shannon kom i sina experiment fram till att entropin för skriven engelska ligger någonstans mellan 0,6 och 1,3 informationsbitar. Metoden bygger – som Shannon själv uttrycker det – på att den som behärskar ett språk också implicit besitter kunskap om statistiska mönster i språket. Denna bygger dels på vår kunskap om ord, grammatik och idiomatiska uttryck, dels på vår kunskap om världen och om vad som är rimligt att säga i olika situationer.

Vad har då detta att göra med stora språkmodeller som GPT-3? Enkelt uttryckt kan man säga att de skapas genom att man vänder på resonemanget bakom Shannons experiment. Shannon utgår i sitt experiment från att den som kan ett språk också kan förutsäga nästa tecken eller ord i en text. När vi tränar stora språkmodeller utgår vi i stället från att en modell som lär sig att förutsäga nästa tecken eller ord i en text implicit också måste förvärva annan kunskap om språket. Stora språkmodeller tränas därför helt enkelt genom att utsättas för Shannons experiment, där de successivt förbättrar sin prediktionsförmåga genom att vi optimerar modellens parametrar för att minimera entropi. För detta behövs ingen lingvistisk annotering, utan bara ren text, där man successivt avslöjar ett tecken eller ett ord i taget för modellen. Detta innebär att det finns nästan

obegränsad tillgång på träningsdata – åtminstone för språk som engelska – och modeller som GPT-3 kan därför tränas på biljontals ord. Genom att samtidigt skala upp modellernas storlek med flera miljarder statistiska parametrar har man lyckats nå en ny nivå i artificiell språkförmåga.

Förutom att dessa resultat är imponerande rent ingenjörsmässigt ger de upphov till ett antal frågor och nya perspektiv beträffande relationen mellan tillämpad AI och teoretisk lingvistik. Om en statistisk modell kan lära sig naturliga språk utan att få explicit information om syntaktisk struktur, betyder det att Chomsky hade fel och att syntaktisk kompetens trots allt inte är en nödvändig komponent i en produktiv språkförmåga? Eller betyder det snarare att modellen faktiskt kan förvärva denna kunskap implicit om den har tillräckligt många parametrar och matas med tillräckligt mycket data? Frågor av den här typen har gett upphov till ett nytt forskningsområde inom AI och språkteknologi, vars mål är att försöka tolka vad de storskaliga modellerna egentligen har lärt sig och att förklara hur denna kunskap kommer till användning när de tolkar och genererar språk. På grund av modellernas storlek och komplexitet måste detta göras med indirekta metoder, antingen genom att man systematiskt manipulerar indata till systemet och observerar vad det får för effekter, eller genom att man analyserar interna representationer i modellerna.

Det intressanta är att vi i båda fallen återigen kan dra nytta av lingvistiskt annoterade data – till exempel trädbanker från UD – för att testa hypoteser om modellernas representationer och kausala mekanismer. Det beskrivs ibland som att lingvistikern ger oss ett mikroskop som vi kan använda för att bättre urskilja detaljerade mönster i de storskaliga modellernas arkitektur. I verkligheten är det förstås inte riktigt så enkelt, och en del av resultaten är svårtolkade och motsägelsefulla, men det är utan tvekan en metod värd att vidareutveckla.

Avslutningsvis kan vi konstatera att kopplingen mellan naturliga språk och artificiell intelligens är starkare än någonsin. Att förstå och producera språk är ett centralt problem inom AI, och teknikutvecklingen inom språkteknologi har även drivit utvecklingen på andra områden, till ex-

empel datorseende. När det gäller kopplingen till lingvistik och traditionell språkforskning är situationen mer oklar. Vi kan använda lingvistiska teorier för att försöka *förstå* de storskaliga modellerna; huruvida vi också kan använda dem för att *förbättra* modellerna är för tillfället en öppen fråga – men sista ordet är definitivt inte sagt ännu.

I en nyligen publicerad enkätundersökning bland språkteknologiforskare – the NLP community metasurvey – svarade ungefär hälften av forskarna att de tror att representationer av språkstruktur baserade på teoretisk lingvistik kommer att vara nödvändiga för att lösa viktiga språkteknologiska problem. Detta kan jämföras med att bara en sjättedel trodde att det enda som behövs för att lösa alla problem är mer data och större modeller. Hur det förhåller sig med den saken kan vi i nuläget bara gissa, men det ska bli mycket spännande att följa den framtida forskningen på området.

Inträdesföredrag den 4 oktober 2022

NOTER

1. Detta krav innebär att tidsåtgången växer som en linjär funktion med meningens längd. En mening med 20 ord får alltså ta dubbelt så lång tid att analysera som en mening med 10 ord, men inte fyra eller åtta gånger så lång tid.
2. För att vara exakt gäller att varje *projektivt* dependensträd skapas av någon sekvens av transitioner. Ett dependensträd är projektivt om det inte innehåller några diskontinuerliga fraser.

REFERENSER

GPT-3 2020. "A robot wrote this entire article. Are you scared yet, human?", *The Guardian* 8/9, <https://www.theguardian.com/commentisfree/2020/sep/08/robot-wrote-this-article-gpt-3>. Hämtad 31/10 2022.

- MICHAEL, JULIAN, ARI HOLTZMAN, ALICIA PARRISH, AARON MUELLER, ALEX WANG, ANGELICA CHEN, DIVYAM MADAN, NIKITA NANGIA, RICHARD YUANZHE PANG, JASON PHANG & SAMUEL R. BOWMAN 2022. "What do NLP researchers believe? Results of the NLP community metasurvey", *arXiv.2208.12852*.
- SHANNON, CLAUDE E. 1951. "Prediction and entropy of printed English", *Bell Systems Technical Journal* 3, s. 50-64.
- TURING, ALAN M. 1950. "Computing machinery and intelligence", *Mind* 59, s. 433-460.