Computers leren taal uit het niets

Mike Kestemont

Menselijke taal begrijpen is notoir moeilijk voor computers: woorden zijn ambigu, dialecten veroorzaken een wilde uitspraakvariatie en veel zinnen zijn moeilijk te vatten zonder een goed begrip van de buitentalige context. Toch is de afgelopen jaren flink vooruitgang geboekt in de taaltechnologie, deels op basis van een methode die 'Deep Learning' heet. Hieronder bied ik een schets van de opgang van deze techniek tegen het achterdoek van de recente geschiedenis van de taaltechnologie, maar ook andere intrigerende toepassingen binnen de informatica. Deep Learning kan tot op zekere hoogte taalkennis verwerven zonder menselijke interventie. Die innovatieve eigenschap creëert interessante perspectieven voor het verbeteren van bestaande taalsoftware.

Taaltechnologie is een vorm van Artificiële Intelligentie, een wetenschapstak waarin men in software het taalvermogen van mensen tracht te imiteren. De taaltechnologie is een nog jonge discipline, maar zij heeft toch reeds een hele weg afgelegd. In een ver verleden was taalsoftware regelgebaseerd en bouwden onderzoekers regels rechtstreeks in systemen in. De computers leerden dus eigenlijk niets zelfstandig en waren vooral uitvoerder van de linguïstische kennis die moeizaam handmatig in regels was geformaliseerd. Onderzoekers werden het echter al snel beu om linguïstische kennis zelf te moeten coderen in duizenden regeltjes, want dat bleek een moeilijke en tijdrovende bezigheid. Op termijn zou een dergelijke aanpak trouwens onhoudbaar blijken: aangezien taal voortdurend verandert, zouden ook de regels voortdurend herzien moeten worden, en daar hadden taaltechnologen weinig zin in.

Deze onvrede vormde de voedingsbodem voor de zogenaamde statistische revolutie in de laatste decennia van de vorige eeuw: in plaats van de oplossing voor bepaalde taaltaken expliciet te programmeren, begon men *Machine Learning* toe te passen. Door het aanreiken van veel voorbeelddata trachtte men een computer autonoom de oplossing voor een bepaald probleem te laten bedenken. Een goed voorbeeld is woordsoortherkenning, waarbij een computer de woorden in een lopende tekst moet categoriseren als werkwoorden, lidwoorden, adjectieven enz. Met voluit geschreven regels duurt het een hele tijd vooraleer je een sluitende oplossing voor dit probleem kan formuleren. Met statistische methodes echter zal een computer al snel uit data leren dat een woord dat volgt op het lidwoord *de*, waarschijnlijk een adjectief of zelfstandig naamwoord zal zijn, althans veeleer dan een werkwoord of nog een lidwoord.

Leren zonder voorbeelden

Een *sine qua non* voor deze vorm van Machine Learning zijn de voorbeelddata, ook wel trainingsmateriaal genoemd: de machine kan immers slechts leren bij de gratie van grote hoeveelheden manueel geannoteerde data. Voor een degelijke woordsoortherkenner heeft men bijvoorbeeld al snel een paar honderdduizenden voorbeelden nodig, waarin door mensen bij elk woord de woordsoort handmatig is aangegeven. Ook in dit geval zijn de inspanning, tijd en kosten die met het aanleggen van dergelijke data gepaard gaan, verre van gering. Bovendien staat taal ook voor deze methodes niet stil: als een taal verandert, zal ook het trainingsmateriaal geactualiseerd moeten worden.

Computers laten trainen op vooraf door mensen geannoteerde data heet ook wel 'gestuurd' leren (*supervised learning*). Hoewel dergelijke methodes een belangrijke vooruitgang hebben gerealiseerd in de taaltechnologie, staat men steeds vaker stil bij de mogelijkheid om computers ook 'ongestuurd' te laten leren (*unsupervised learning*), en dus niet langer op basis van data die eerst door mensen werden geannoteerd. Een dergelijke aanpak is natuurlijk ambitieus, maar minder gek dan op het eerste gezicht zou lijken. Ook kinderen verwerven immers taal zonder al te veel expliciete sturing: gewoon door het 'ervaren' van taal kunnen kinderen blijkbaar genoeg kennis distilleren om deze taal op den duur zelf te gaan produceren.

Deep Learning

Een belangrijke recente ontwikkeling in het gebied van het ongesuperviseerd leren betreft *Deep Learning*: het gaat om een paradigma in de informatica waarin algoritmes worden ontwikkeld die inderdaad getraind kunnen worden op ongeannoteerde data. Deze algoritmes nemen vaak de vorm aan van neurale netwerken: een softwarearchitectuur waarin informatie-eenheden zijn gestructureerd in hiërarchische lagen die steeds bovenop elkaar worden gebouwd. Het belang van een dergelijke, gelaagde of 'diepe' architectuur wordt doorgaans geïllustreerd aan de hand van een voorbeeld uit de beeldverwerking of *computer vision*, het domein waarin *Deep Learning* de vroegste successen heeft geboekt. De bekendste applicatie van het zogenaamde 'diep leren' is momenteel gezichtsherkenning in digitale foto's: gegeven het raster van pixeltjes dat een foto met een gezicht voorstelt, is de taak van het netwerk om te voorspellen om welk individu het precies gaat. Dat lijkt een bijzonder moeilijke taak, maar op sociale

netwerksites als *Facebook* hebben we allemaal al kunnen ervaren hoe verrassend (en soms akelig ...) accuraat dergelijke algoritmes werken.

Wanneer een gelaagd neuraal netwerk wordt getraind op dergelijke pixelrasters, heeft men gemerkt dat de verschillende lagen een heel ander type informatie capteren. De eerste lagen in het netwerk nemen heel primitieve vormen waar, zoals felle lokale contrasten. Pas in de diepere lagen in het netwerk worden deze primitieve vormen samen gepuzzeld tot grotere en steeds abstractere vormen, zoals oren en ogen. Uiteindelijk, in de diepste en meest complexe lagen, slaagt het netwerk erin de hele gezichten van individuen te detecteren. Interessant genoeg is aangetoond dat visuele perceptie bij zoogdieren op een vergelijkbare manier werkt: als mensen hun omgeving waarnemen, zullen ook zij eerst heel primitieve, lokale vormen waarnemen en die vervolgens, dieper in het brein, tot meer complexe patronen combineren. Daarom wordt vaak gezegd dat Deep Learning geïnspireerd is op het menselijke brein – al is er ook voldoende onderzoek dat aantoont dat we deze analogie niet te ver mogen doortrekken.

Deep Learning is in veel opzichten slechts een *buzz word*, dat ook makkelijk door de media wordt opgepikt. Een meer neutrale benaming is *representation learning*, omdat veel van deze algoritmes niet zozeer bezig zijn met het oplossen van een probleem, maar wel met hoe data het best gerepresenteerd kunnen worden om een bepaald probleem op te lossen. Bijvoorbeeld: bij gezichtsherkenning is het grootste deel van het netwerk eigenlijk bezig met het leren van hiërarchische representaties van foto's. Zodra deze goed geleerd zijn, is de eigenlijke identificatie van een gezicht kinderspel geworden.

Deep Learning heeft de laatste tijd tot belangrijke doorbraken geleid in verschillende wetenschapstakken. Dat drie tenoren uit het vakgebied recent een overzichtspaper mochten publiceren in het hoog aangeschreven vakblad *Nature* spreekt in dat opzicht overigens ook boekdelen. Binnen beeldverwerking was het tot voor kort moeilijk om op een foto de aap van de banaan te onderscheiden: nu zijn computers zelfs in staat om verschillende hondenrassen te onderscheiden op foto's. Die hoge vlucht is de *Silicon Valley* niet ontgaan: giganten als *Google* of *Facebook* investeren massaal in deze technologie, zelfs in die mate dat onderzoekers uit deze bedrijven bepaalde wetenschapsdomeinen zijn gaan domineren.

Betekenis uit context

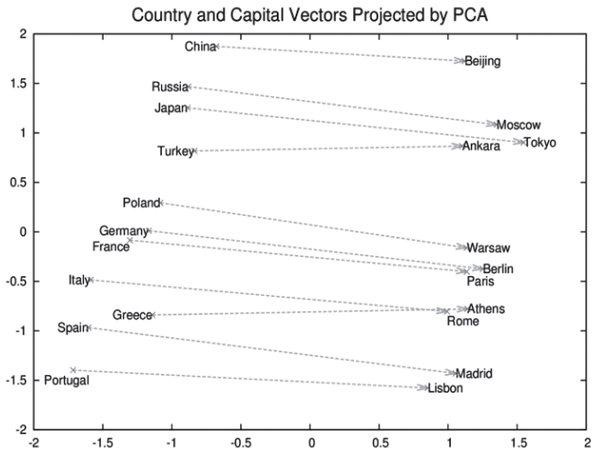
Terug naar taal. In de laatste jaren zijn onderzoekers ook actief bezig met de transfer van *Deep Learning*-technieken naar de taaltechnologie. Vooral de ontwikkelingen binnen het ongestuurd leren van manieren om woorden te representeren, is daarbij erg vruchtbaar gebleken. Deze ontwikkelingen situeren zich in het domein van de semantiek of betekenisleer, meer bepaald de 'distributionele semantiek'. Het basisprincipe van de distributionele semantiek is, opnieuw, al decennia geleden geformuleerd door mensen als Harris (1954) en Firth (1957): het centrale idee is dat woorden op zich vrij weinig betekenis dragen, maar vooral betekenis ontlenen aan de context waarin ze optreden. In menselijke taal is het namelijk zo dat woorden met een gelijkaardige semantiek in dezelfde contexten zullen optreden. Hoewel **blarf* een onbestaand woord is, bijvoorbeeld, suggereren volgende zinnen heel sterk dat het om een soort tafel moet gaan: 'Het eten staat op de *blarf', 'We kopen een nieuwe *blarf voor in de keuken'. Uit de context waarin woorden worden gebruikt, kunnen we dus hun betekenis afleiden – of in de woorden van Firth: 'You shall know a word by the company it keeps'.

In de taaltechnologie is een goed begrip van woordbetekenissen cruciaal: als een gebruiker in een zoekscherm gebruik maakt van een specifieke term, is het bijvoorbeeld belangrijk dat we ook documenten met evidente synoniemen kunnen retourneren. Hoewel het onderzoek naar computationele modellen in dit gebied al langer gaande is, is er recent sprake van een stroomversnelling. In 2013 publiceerden Tomas Mikolov et al. een invloedrijke paper waarin zij een nieuwe methode beschreven om woordrepresentaties te extraheren uit grote hoeveelheden tekst, dit op basis van technieken uit de *Deep Learning*. Een aantal innovaties zijn belangrijk hier. In de eerste plaats heeft deze methode geen noemenswaardige menselijke supervisie nodig: de techniek heeft genoeg aan het bestuderen van welke woorden vaak samen optreden in grote tekstverzamelingen. Bovendien is het algoritme veel simpeler dan eerdere benaderingen: daardoor kan het op veel grotere datasets gedraaid worden en dus uit veel meer teksten leren dan vroeger.

Mikolovs methode werd 'word2vec' gedoopt, omdat de methode uiteindelijk voor elk woord een 'vector' leert: een relatief kleine reeks getallen (bv. 300) die gebruikt kan worden om het woord voor te stellen. De idee is natuurlijk dat woorden met een vergelijkbare semantiek ook gelijkaardige vectoren krijgen toebedeeld. Dergelijke representaties worden ook wel *embeddings* genoemd, omdat ze modelleren in welk soort talige context woorden typisch zijn 'ingebed'. Om na te gaan of het model goede embeddings heeft geleerd, zijn verschillende testen gangbaar in het vak. Zo kunnen we het model vragen welke 'dichtste buren' een bepaald woord heeft. Mikolovs model zou bijvoorbeeld voor het woord *france*, de woorden *spain*, *belgium* en *netherlands* als buren teruggeven – waaruit duidelijk blijkt dat het model heeft geleerd dat landsnamen een vergelijkbare vector moeten krijgen. Mikolov et al. hebben hier een bijzonder interessante test aan toegevoegd: het onderzoeksteam heeft namelijk opgemerkt dat het *word2vec*-model redelijk complexe analogieën kon oplossen door het gebruik van verrassend simpele rekensommen. De bewerking *king-man+woman* leverde bijvoorbeeld het woord *queen* op. Het denkprobleem dat de computer hier moet oplossen is: 'Welk woord verhoudt zich tot *vrouw*, zoals *koning* zich verhoudt tot *man*?' Ook

intrigerende regelmatigheden als de volgende bleken werkzaam (zie ook afbeelding 1):

new_york-pizza+japan ~ sushi
paris-france+italy ~ rome
quickly-quick+slow ~ slowly
write-writes+play ~ plays



Afbeelding 1

[http://google-opensource.blogspot.de/2013/08/learning-meaning-behind-words.html]

Aan deze verrassende uitkomsten vallen twee dingen op: de gevonden regelmatigheden zijn gevoelig voor meer dan enkel semantiek, want blijkbaar kunnen ook courante woordvervoegingen op deze manier worden gemodelleerd. Nog opvallender is wellicht dat dit model over een soort 'gezond verstand' (*common sense*) bleek te beschikken. Dergelijke, alledaagse kennis (bv. 'verse koffie is heet') wordt meestal afgezet tegenover encyclopedische kennis ('arabica en robusta zijn soorten koffiebonen'). Encyclopedische kennis kan een computer snel opdoen door bijvoorbeeld Wikipedia af te struinen; echter, de simpele observatie dat verse koffie heet wordt opgediend, is zo evident dat dit nergens expliciet wordt opgeschreven. Toch bleek *word2vec* vormen van dergelijke *common sense* te hebben verworven, louter dus, door veel tekst te lezen.

Geheime saus

Zonder onderdrijving kan men stellen dat het *word2vec*-onderzoek is ingeslagen als een bom in de computerlinguïstiek – geholpen door het feit dat *Google* een efficiënte implementatie van het algoritme heeft vrijgegeven. Mikolovs vectoren bleken bijvoorbeeld bijzonder nuttig als aanvulling op bestaande software voor taaltechnologie. Voor woordsoortherkenning blijken *embeddings* bijvoorbeeld heel nuttig, want adjectieven zullen nu eenmaal een ander type *embedding* hebben dan pakweg lidwoorden. Ook voor automatische vertaling zijn de mogelijkheden legio. Het woord *hond* heeft nu eenmaal een vergelijkbare inbedding in het Chinees als in het Hongaars, zodat deze correspondenties makkelijker kunnen worden opgespoord. Dat deze representaties moeiteloos kunnen worden geoogst uit ongeannoteerde corpora draagt natuurlijk bij tot hun populariteit. Het is zelfs wel gesteld dat de *word2vec*-vectoren een 'geheime saus' vormen, waarmee veel hedendaagse taalsoftware wordt overgoten.

Toch zijn enkele kanttekeningen nodig. De buitenproportionele aandacht voor *word2vec* doet bijvoorbeeld geen recht aan het waardevolle onderzoek dat al vroeger was uitgevoerd op dit gebied. Recent werk heeft trouwens aangetoond dat ook traditionele modellen Mikolovs koningin-analogieën kunnen oplossen, als ze maar op genoeg data worden getraind. Of *word2vec* ook echt een 'diep' model mag heten, is trouwens tevens voor discussie vatbaar. Het oorspronkelijke model heeft eigenlijk slechts één laagje, in tegenstelling tot de tientallen lagen die in beeldverwerking gangbaar zijn. Ook werpen ervaren rotten in het vak de vraag op of dit allemaal wel zo nieuw is: neurale netwerken draaien al een tijd mee in het vak, en deze (tijdelijke?) heropleving mag het onderzoek naar andere methodes niet doen verwateren. Toch is het een belangrijke verdienste dat dit onderzoek de computationele studie van de semantiek in een stroomversnelling heeft gebracht, en ook de mogelijkheid tot het ongestuurd leren van taal duidelijker op de onderzoeksagenda heeft geplaatst. Duidelijk is alleszins dat de informatica, vandaag meer dan ooit, interesse toont voor taal. En meer interesse voor taal, dat kan op termijn enkel een goede zaak zijn.

Referenties

Firth, J.R. (1957), 'A synopsis of linguistic theory 1930-1955'. In: *Studies in Linguistic Analysis,* blz. 1–32.

Harris, Z. (1954), 'Distributional structure'. In: *Word* 10, blz. 146–162.

LeCun, Y., Bengio, Y. en Hinton, G. (2015), 'Deep Learning'. In: *Nature* 52(1), blz. 436–444.

Manning, C.D. (te verschijnen), 'Computational Linguistics and Deep Learning'. In: *Computional Linguistics*.

Mikolov, Tomas, Ilya Sutskever, Kai Chen, Greg S. Corrado & Jeffrey Dean (2013), 'Distributed representations of words and phrases and their compositionality'. In: C. J. C. Burges, L. Bottou, M. Welling, Z. Ghahramani en K. Q. Weinberger (red.), *Advances in Neural Information Processing Systems 26 (NIPS 2013)*. Curran, blz. 3111–3119.

Mike Kestemont is als docent 'Digitale tekstanalyse' verbonden aan het departement Letterkunde van de Universiteit Antwerpen. Hij wordt momenteel sterk geboeid door de plotse opkomst van het fenomeen 'Deep Learning' en de waardevolle applicaties van deze leermethodes in de geesteswetenschappen.

e-mail: mike.kestemont@uantwerpen.be