

Lang leve neologismen!

Wat bepaalt de levensduur van
neologismen? Een corpusonderzoek.

Ella Van Lint

Masterproef aangeboden binnen de opleiding
Master in de Taal- en Letterkunde

Promotor: dr. Kris Heylen

Academiejaar 2021-2022

102.958 tekens



Ik verklaar me akkoord met de code of conduct van de faculteit Letteren
voor geloofwaardig auteurschap.

Abstract

Studies on lexical innovation and neologisms traditionally use frequency development over time as a starting point to investigate their life cycles. These studies often focus on individual new words and pay little attention to the influence of certain characteristics on the success of neologisms on the aggregate level. They also tend to exclude short-lived “unsuccessful” lexical innovations from their investigations. Studies in the field of evolutionary linguistics on the other hand are situated on an aggregate level but take a macrodiachronic approach (multiple centuries) and do not generally deal with neologisms. Therefore, an aggregated, microdiachronic overview (decades) of the life course and survival rate of neologisms, and what characteristics ensure success, is a gap in research on neologisms, especially in Dutch. This study aims to fill that lacuna.

Two types of sources were used to answer this question. Firstly, a lexicographically validated list of neologisms was obtained by combining a Dutch dictionary of new words (WNW - *Woordenboek van Nieuwe Woorden*) and a database of candidate neologisms (*Neoloog*), which resulted in 3804 neologisms “born” between 2000 and 2021. Secondly, data about their attestation and dating were obtained from a Dutch monitor corpus (*Corpus Hedendaags Nederlands*). This study aimed to identify the characteristics that play a role in the success and life span of neologisms through applying survival analyses. The characteristics that were thought to have an influence are: source (WNW vs. *Neoloog*), length, Levenshtein distance, article, gender, part of speech, word formation (compounds), productivity, origin (loanwords vs. native words), semantic domain, and synonymy. These data were then analyzed through Kaplan-Meier curves and Cox Proportional Hazard models, which are two statistical methods to investigate survival rates, in this case, applied to new words.

The results suggest that short words which become diffused over two varieties of Dutch - especially Dutch-Dutch to Belgian-Dutch - and which are included in the WNW have the highest survival rates. The WNW lexicographers thus use relevant inclusion criteria, and their expertise contributes something yet unknown to the survival rate of neologisms. Future studies might want to investigate this unknown variable. A

hypothesis is that "origin" (loanwords vs. native words) could be the missing link. It was also concluded that patterns that occur on a macrodiachronic level appear in microdiachrony as well, which suggests that survival analyses are a valuable tool to study the success of neologisms in addition to traditional time-frequency graphs.

Voorwoord

In eerste instantie bedank ik mijn promotor, dr. Kris Heylen, om altijd even enthousiast en rustig tegelijk tijd vrij te maken om vragen te beantwoorden, feedback te geven of te helpen bij het coderen. Zonder zijn hulp was ik niet veel verder gekomen dan "print("Hello, world")". Verder ook dank aan het Instituut voor de Nederlandse Taal, waarmee deze masterproef in samenwerking is, om de nodige data en technologische infrastructuur te voorzien om dit onderzoek te verwezenlijken. Ook wil ik graag mijn ouders, mijn broer en Yentl Cresens bedanken voor de steun en om af en toe stukjes tekst na te lezen. Ten slotte bedankt aan alle taalcreatievelingen die geweldige neologismen uitvinden. Mijn favorieten blijven *jadicalisejing*, *adoptiekoe* en *koekoekwonen*.

Inhoudstabel

1. Inleiding.....	1
2. Literatuurstudie.....	4
2.1. Detectie en observatie.....	4
2.2. Descriptief perspectief.....	6
2.3. Lexicografisch perspectief.....	9
2.4. Sociolinguïstisch en cognitief-linguïstisch perspectief.....	12
2.5. Evolutionair-linguïstisch perspectief.....	14
2.6. Vraagstelling.....	16
3. Methodologie.....	17
3.1. Data.....	17
3.1.1. Woordenboek van Nieuwe Woorden (WNW).....	17
3.1.2. Neoloog.....	18
3.2. Afhankelijke variabelen: Corpus Hedendaags Nederlands.....	18
3.3. Dataverwerking en -selectie.....	19
3.4. Onafhankelijke variabelen.....	21
3.4.1. Fonetische kenmerken.....	21
3.4.2. Morfologische kenmerken.....	23
3.4.3. Sociolinguïstische en cognitief-linguïstische kenmerken.....	26
3.4.4. Semantische kenmerken.....	28
4. Beschrijvende analyse van enkele frequentieverlopen.....	30
5. Resultaten van de survivalanalyses.....	41
5.1. Kaplan-Meier.....	41
5.1.1. Fonetische kenmerken.....	43
5.1.2. Morfologische kenmerken.....	47
5.1.3. Sociolinguïstische en cognitief-linguïstische kenmerken.....	52

5.1.4. Semantische kenmerken	54
5.2. Cox Proportional Hazards model	57
6. Conclusies	61
7. Bibliografie	64
8. Appendix.....	69
Appendix I: Neologismen uit het WNW	69
Appendix II: Neologismen uit Neoloog.....	80
Appendix III: Pythonscripts	96
Appendix IV: Survivaltabellen, chi-squares, p-waarden en pairwise comparisons	133
Appendix V: Cox Proportional Hazards modellen	174

Lijst van figuren en tabellen

Figuur 1: Aantal neologismen dat opgenomen is per jaar (N=3804).....	30
Figuur 2: Overlevingsduur van alle neologismen (N=3804), waarbij 0,1 staat voor minder dan 1 jaar oud.	31
Figuur 3: Aantal jaar tot neologismen hun piek bereiken (N=2299).	32
Figuur 4: Hoeveel jaar na hun ontstaan bereiken neologismen van minstens 5 jaar oud (N=1605) hun piek?.....	33
Figuur 5: Continuïteit van observatie (N=2085) & Verschillende graden van continuïteit.....	34
Figuur 6: Hoogfrequente neologismen.	36
Figuur 7: Middelfrequente neologismen.	36
Figuur 8: Laagfrequente neologismen.....	37
Figuur 9: Verscheidenheid in frequentiepieken.....	37
Figuur 10: Verschil in frequentieverloop tussen BN en NN.....	38
Figuur 11: Frequentieverloop van 9 frequente leenwoorden (zwart) vs. 10 frequente inheemse neologismen (grijs). De stippellijnen zijn de gemiddelde relatieve frequentie.....	39
Figuur 12: Algemene survivalcurve (N=3804).....	42
Figuur 13: Survivalcurve met als verklarende variabele "bron".	43
Figuur 14: Survivalcurve met als verklarende variabele "lengte".....	44
Figuur 15: Survivalcurve met als verklarende variabele "Levenshteinafstand".....	45
Figuur 16: Survivalcurve met als verklarende variabele "gewogen Levenshteinafstand".....	46
Figuur 17: Survivalcurve met als verklarende variabele "frequentie van het dichtstbijzijnde woord".	47
Figuur 18: Survivalcurve met als verklarende variabele "lidwoord".....	48
Figuur 19: Survivalcurve met als verklarende variabele "geslacht".....	49
Figuur 20: Survivalcurve met als verklarende variabele "woordsoort".	50
Figuur 21: Survivalcurve met als verklarende variabele "samenstelling".	51
Figuur 22: Survivalcurve met als verklarende variabele "productiviteit van de samenstellende delen".	52
Figuur 23: Survivalcurve met als verklarende variabele "diffusie".....	53

Figuur 24: Survivalcurve met als verklarende variabele "herkomst".	54
Figuur 25: Survivalcurve met als verklarende variabele "een of meer domeinen"	55
Figuur 26: Survivalcurve met als verklarende variabele "automatisering, informatie en communicatie".	55
Figuur 27: Survivalcurve met als verklarende variabele "synonymie".	56
Tabel 1: Cox Proportional Hazards model met alle variabelen.....	58
Tabel 2: Cox Proportional Hazards model met variabelen "bron", "diffusie" en "aantal karakters".....	60

1. Inleiding

Het woord *neologisme* komt van het Griekse *νέο- néo-*, "nieuw" en *λόγος lógos*, "woord". Een neologisme is dus een "nieuw woord". Volgens het Instituut voor de Nederlandse Taal komen er "ieder jaar [...] duizenden nieuwe woorden, neologismen, bij" (WNW 2022). Ook wordt er op het einde van het jaar telkens het woord van het jaar gekozen, maar laat ons eerlijk zijn, wie gebruikt er in zijn dagelijks taalgebruik nog het woord *samsonseks* uit 2016? De meeste van zulke neologismen zijn namelijk modegrillen en eendagsvliegen. Toch zijn er ook nieuwe woorden die wel lange tijd meegaan. Zo is het woord van het jaar 2013 *selfie* bijna niet meer weg te slaan uit ons taalgebruik. De vraag die dan rijst is: wat bepaalt het succes en de levensduur van een neologisme? Zijn er bepaalde kenmerken die zorgen voor een hogere overlevingskans?

Het bekendste antwoord op die vraag zijn de FUDGE-criteria van Metcalf (2002, cf. 2.3.). Volgens hem zijn frequentie; onopvallendheid; diversiteit; het genereren van vormen en betekenissen; en duurzaamheid van het benoemde concept de belangrijkste kenmerken op basis waarvan men kan voorspellen of een nieuw woord een "blijvertje" is. Vooral aan frequentie(verlopen) wordt in traditioneel neologismenonderzoek veel aandacht geschonken. Zo vergelijkt Renouf (2013, cf. 2.2.) de levensloop van woorden met die van een dier. Dat doet ze aan de hand van frequentieverlopen, grafieken waarop te zien is hoe een dier of woord geboren wordt, stijgt in aantal, weer afneemt en eventueel sterft. Een voorwaarde voor die manier van werken is natuurlijk dat er een levensloop is, wat vaak niet het geval is voor kortlevende neologismen. Traditioneel neologismenonderzoek behoudt dan ook vaak niet die kortlevende, "onsuccesvolle" woorden en bestudeert individuele woorden op een microdiachroon niveau (enkele jaren).

In evolutionair-linguïstisch onderzoek van het lexicon bestaat er een geaggregeerde insteek, maar die bevindt zich op macrodiachroon niveau (meerdere eeuwen) en houdt zich meestal niet bezig met neologismen. Zulk onderzoek gebruikt survivalanalyses om de overlevingskans van woorden in te schatten. In tegenstelling tot frequentieverlopen houden survivalanalyses rekening met de woorden die tijdens de onderzochte periode sterven. Uit zulk onderzoek van Van de Velde &

Keersmaekers (2020) en Van de Velde (2020) bleek al dat korte, frequente, unieke nomina en verba een hoge overlevingskans hebben (cf. 2.5.). De dataset van die studies verschilt echter met de data van dit onderzoek. Bij Van de Velde & Keersmaekers gaat het bijvoorbeeld om woorden uit het Grieks die al geëtableerd zijn; in dit onderzoek staat nieuwe woordenschat centraal. Woorden die al langer in het lexicon zitten, overleven in het begin langer, in tegenstelling tot neologismen die pasgeboren zijn en massale "kindersterfte" ondergaan. De evolutionaire taalkunde focust ook typisch op tijdspannen van meerdere eeuwen, terwijl neologismen vaak maar een dag tot een paar decennia leven. Een bijkomende vraag is of de survivalanalyses patronen aan het licht brengen op microdiachronie, die ook op macrodiachronie voorkomen.

Het innovatieve karakter van dit onderzoek is dus tweeledig: enerzijds vertrekt traditioneel neologismenonderzoek van individuele frequentieverlopen en houdt het geen rekening met kortlevende nieuwe woorden, terwijl ik met geaggregeerde survivalanalyses zal werken die net wel rekening houden met verschillen in levensduur. Anderzijds worden de survivalanalyses in de evolutionaire taalkunde typisch op macrodiachrone schaal toegepast, terwijl ik ze op microdiachrone schaal zal toepassen. Het gebruik van survivalanalyses voor microdiachrone verschijnselen zoals neologie is dus een lacune in de lexicologie in het Nederlands. Het doel van deze thesis is om dat gat op te vullen en aan de hand van survivalanalyses te onderzoeken welke kenmerken zorgen voor een hogere overlevingskans bij neologismen.

Om dat te verwezenlijken werd een lexicografisch gevalideerde lijst van neologismen samengesteld met nieuwe woorden "geboren" tussen 2000 en 2021 uit het *Woordenboek van Nieuwe Woorden* en een databank met kandidaat-neologismen *Neoloog*. Ook de kenmerken die het succes van de neologismen kunnen verklaren werden uit het WNW gehaald. Informatie over hun eerste en laatste observatie werd opgehaald uit het *Corpus Hedendaags Nederlands*. Aan de hand van die informatie kon voor elke verklarende variabele een survivalanalyse gemaakt worden. Die verklarende variabelen zijn de kenmerken die voortkomen uit de literatuurstudie (hoofdstuk 3): bron (WNW vs. *Neoloog*), lengte, Levenshteinafstand, lidwoord, geslacht, woordsoort, woordvorming (samenstelling), productiviteit, diffusie, herkomst (leenwoorden vs. inheemse woorden), semantisch domein en synonymie.

De opbouw van de thesis is als volgt: in hoofdstuk 2 wordt een overzicht gegeven van onderzoek naar neologismen, opgedeeld in verschillende perspectieven. Daaruit volgen de variabelen die onderzocht zullen worden, alsook specifieke hypothesen en vragen waarop deze thesis een antwoord zal proberen te formuleren. De variabelen en de neologismenlijst waarop dit onderzoek gebaseerd is worden voorgesteld in hoofdstuk 3. Een korte analyse van die data op basis van frequentie(verlopen) volgt in hoofdstuk 4. In hoofdstuk 5 worden de survivalanalyses uitgevoerd op de variabelen die voortkomen uit de literatuurstudie. Hoofdstuk 6 brengt alles samen in een conclusie en doet suggesties voor vervolgonderzoek.

2. Literatuurstudie

Onderzoek naar neologismen omvat een veelheid aan perspectieven: detectief en observatief, descriptief, lexicografisch, cognitief-linguïstisch, sociolinguïstisch, attitudineel, evolutionair-linguïstisch... In wat volgt geef ik een overzicht van hoe die invalshoeken toegepast worden of kunnen worden op de studie van neologismen. In de realiteit overlappen die perspectieven elkaar echter en zijn ze hier dus artificieel opgesplitst met het oog op overzichtelijkheid. Hoe deze thesis de traditionele en minder traditionele perspectieven en methoden toepast en combineert wordt ook besproken.

2.1. Detectie en observatie

In recente jaren is de detectie en observatie van nieuwe woorden aanzienlijk makkelijker geworden. Waar onderzoekers vroeger manueel in de krant of magazines nieuwe woorden moesten zoeken, voeren computationele modellen ontwikkeld voor en/of door de linguïstiek datzelfde proces nu automatisch uit. Menselijke *word-watchers* zijn immers onderworpen aan subjectiviteit en onvolledigheid. Computationeel-linguïstisch onderzoek naar neologismen wordt doorgaans in twee stappen uitgevoerd: detectie en observatie.

De detectie van neologismen is pas de laatste jaren op gang gekomen onder invloed van taalveranderingsonderzoek dat grote diachrone corpora gebruikt. De opkomst van zulke corpora heeft het tracken van neologismen erg bevorderd (Cartier 2017). Enkele tools die gebruik maken van zulke corpora zijn *NeoCrawler* (Kerremans et al. 2012), *NeoTrack* (Janssen 2008), *ZeitGeist* (Veale 2006), *NeoVeille* (Cartier 2017) en *Language Monitor* (Kosem et al. 2021). Het detectiegedeelte gebeurt vaak aan de hand van twee methoden: exclusielijsten en *machine learning* (Falk 2018). Voor de observatie maakt men gebruik van NLP en databases om het verloop en gebruik van neologismen te onderzoeken. Daarvoor heeft men ook een monitorcorpus met timestamps nodig om de attestaties te kunnen volgen in de tijd. Een overzicht van enkele computationele neologismentrackers geeft een zicht op hoe detectie en observatie in de praktijk ingezet worden.

Ten eerste is er voor het Engels *NeoCrawler* (2012), bestaande uit de *Discoverer* en de *Observer*. De *Discoverer* identificeert neologismen op websites verspreid over het hele internet op basis van een *string matching* algoritme dat nakijkt of een token in het referentiecorpus zit. Het referentiecorpus bestaat uit teksten van *Wikipedia* en zorgt ervoor dat bestaande woorden niet als neologismen worden aanzien. Het fungeert dus als exclusielijst. Daarna volgt manuele *post-processing* om kandidaat-neologismen eruit te filteren. De *Discoverer* identificeert enkel morfologische en formele nieuwigheid. De *Observer* verzamelt dan via *Google API* nieuwe voorkomens van het neologisme, downloadt ze, stript ze van onnodige informatie en plaatst ze in een database die gebruikt kan worden voor onderzoek. *NeoCrawler* is dus een semiautomatische identificatie- en observatietool voor neologismen in het Engels (voor meer info, zie Kerremans et al. 2012; Kerremans & Prokić 2018).

Ook in het Frans bestaat er een semiautomatische tool voor het detecteren en documenteren van neologismen: *Logoscope* (2018). Die tool verzamelt nieuwe woorden uit online kranten zoals *Le Monde* en *Le Figaro*. Ook hier wordt een exclusielijst gebruikt bestaande uit Franse woorden uit lexicons en corpora. Via *machine learning* worden de meest plausibele kandidaat-neologismen bovenaan de lijst gezet. Daarna wordt manueel besloten of het effectief om een nieuw woord gaat. Ook informatie over het woord wordt gedocumenteerd, afhankelijk van het type van informatie gebeurt dat manueel of automatisch. *Logoscope* combineert dus de twee meest gangbare methoden voor neologismedetectie: exclusielijsten en *machine learning* (voor meer info, zie Falk et al. 2018).

Een tool die zowel detectie als observatie voor zeven talen (Frans, Chinees, Grieks, Italiaans, Pools, Portugees, Russisch, Tsjechisch en Spaans) automatiseert is *Neoveille* (2018). Verschillende RSS-feeds worden dagelijks gescand op neologismen. Geavanceerde data-analyse laat toe de levenscyclus van neologismen te visualiseren dankzij een combinatie van metadata en *text mining*. Ook formele informatie over de neologismen wordt toegevoegd via *machine learning*. In tegenstelling tot *NeoCrawler* kan *Neoveille* niet enkel formele neologismen, maar ook semantische neologismen opsporen (voor meer info, zie Cartier 2017). Semantische neologie zorgt namelijk voor enkele uitdagingen. Er moet rekening gehouden worden met de context aangezien semantiek verder gaat dan enkel woordniveau. Enkele

projecten zoals *AVIATOR* (Renouf 1993), *WebCorpLSE* (Kehoe & Gee 2009; Renouf 2009) en onderzoek in het Deens (Nimb et al. 2020) en het Sloveens (Pollak et al. 2019) wijzen daarvoor op het nut van N-grams en collocaties (Kosem et al. 2021). Semantische neologie valt echter buiten het bestek van dit onderzoek (cf. 2.2.).

Detectie en observatie van neologismen is dus een complex proces dat veel computationele implementaties vraagt. De detectie voor dit onderzoek heb ik bewust overgelaten aan ervaren lexicografen: de onderzochte neologismen komen uit het *Woordenboek Nieuwe Woorden* (WNW) en neologismendatabank *Neoloog* (cf. 3.1.). Zoals eerder vermeld is er voor de observatie van neologismen steeds nood aan een monitorcorpus met timestamps. De observatie voor deze thesis zal dus uitgevoerd worden via een API die de voorkomens van de neologismen uit het Corpus Hedendaags Nederlands (CHN) haalt. In dat corpus kunnen de dateringen van de attestaties makkelijk achterhaald worden. Die resultaten worden dan gevisualiseerd door middel van een combinatie van frequentieverlopen en survivalanalyses.

2.2. Descriptief perspectief

Dankzij de ontwikkeling van methoden voor detectie en observatie kan er ook op een descriptieve manier gekeken worden naar de levensloop van neologismen. Dat soort onderzoek vertrekt van frequenties en visualiseert deze in een grafiek of staafdiagram. Zo kan men een levenscyclus opstellen. Die levenscyclus wordt in de literatuur vergeleken met die van een levend wezen (Renouf 2013). Een bepaald dier wordt geboren, groeit op, vermenigvuldigt zich - het ene dier al wat meer dan het andere - en sterft uiteindelijk. In sommige gevallen overlijdt het laatste exemplaar en dan kunnen we de diersoort uitgestorven verklaren. Hetzelfde geldt voor woorden.

In descriptief onderzoek van Renouf (2013) wordt de levenscyclus van enkele neologismen onderzocht aan de hand van frequentieverlopen. Ze deelt het leven van een neologisme in in acht fasen: geboorte, stijgende frequentie, orthografische aanpassing, productiviteit, creativiteit, settelen, veroudering, dood (met eventueel ook nog semantische neologie en wedergeboorte). Ik zal elke fase hieronder kort bespreken.

Geboorte. Elk woord is op een bepaald moment en op een bepaalde manier ontstaan. Neologismen zijn vaak het gevolg van samenstelling (*adoniscomplex*) of

derivatie (*appen*) en zelden van totaal arbitraire uitvinding (voor de verschillende types neologismen zie 3.4.).

Stijgende frequentie. Na de geboorte kan het alle kanten uitgaan met een neologisme. Sommige nieuwe woorden stijgen plots in frequentie doordat ze populair zijn, andere stijgen gradueel, dalen of stagneren. Pieken, dalen en plateaus kunnen elkaar afwisselen doorheen de tijd.

Orthografische aanpassing. Naarmate een woord vaker voorkomt, is er nood aan standaardisatie. Soms wordt een woord geboren en gespeld met een koppelteken, zoals *deel-economie*, waarna de spelling veranderd wordt naar *deeleconomie*, in lijn met de Nederlandse spellingsregels. Ook hoofdletters worden veranderd naar kleine letters, of omgekeerd, en bij afkortingen worden punten toegevoegd of weggelaten.

Productiviteit. Om verder in de analogie te blijven van de levenscyclus van een dier, kunnen we de productiviteit van een woord vergelijken met voortplanting. Net zoals dieren jongen kunnen krijgen, kan een woord zich "voortplanten" door samenstellingen of derivaties van zichzelf te maken. Sommige woorden zijn zoals panda's en brengen nauwelijks kinderen voort, terwijl andere kweken als konijnen. Een voorbeeld van zo'n erg productief neologisme is *alt-rechts*, waaruit *alt-rechtsbeweging*, *alt-rechtsgroepering*, *alt-rechtslid*... zijn ontstaan. Een ander voorbeeld van productiviteit is derivatie, zoals *blogger*. Een niveau hoger in de stamboommetafoor zijn er ook de ouders van het dier, of in het geval van samenstellende neologismen de samenstellende delen, bijvoorbeeld *adoptiekoe* bestaat uit *adoptie* + *koe*, waaruit *adoptieouders* of *koemelk* ontstaan zijn. De productiviteit van de samenstellende delen kan een rol spelen in het succes van neologismen en zal in dit onderzoek dus ook onderzocht worden (cf. 3.4.2.).

Creativiteit. Neologismen kunnen ook creatief gebruikt worden in een woordspel of *pun*. Meerwoordige neologismen lenen zich daar het beste toe, omdat ze genoeg lexicaal materiaal hebben om mee te spelen (Waszink 2016).

Settelen. Na de geboorte en eventuele stijging in frequentie, daalt het gebruik van een neologisme en kan het stranden in een plateau. Het woord wordt dan wel nog gebruikt, maar minder frequent dan in het begin en met minder pieken en dalen.

Veroudering (in het Engels *obsolescence*). Men spreekt van veroudering wanneer een woord een erg lage frequentie heeft en de curve zich in dalende trend voortzet.

Dood. In tegenstelling tot dieren, bestaat er voor woorden geen consensus over wat nu precies "dood" betekent. Is een woord "dood" als het een eeuw niet meer is voorgekomen, of enkele jaren? Voor Renouf (2013) is een woord "overleden" als het nog zo weinig voorkomt dat het eigenlijk als "dood" kan beschouwd worden. De dood van een neologisme is een erg belangrijk gegeven voor dit onderzoek, aangezien we willen weten welke kenmerken zorgen voor een vroegere of latere dood. In deze thesis is ervoor gekozen om een neologisme "dood" te verklaren na de laatste datering, ongeacht hoeveel het daarvoor voorkwam in het corpus (cf. 4., p. 35).

Semantische neologie. Een ander proces waaruit neologismen ontstaan kunnen zijn is semantische neologie. Zo diende het woord *slim* eerst als aanduiding voor intelligentie bij mensen, maar wordt het nu onder invloed van het Engelse *smart* ook voor technologische snufjes gebruikt (Waszink 2016), zoals een *slimme horloge*. In dat geval spreken we van neosemie of semantische neologie. Het woord *slim* wordt in beide betekenissen nog gebruikt, maar enkel de tweede betekenis is een neologisme. Het is echter moeilijk om manueel die betekenissen uit elkaar te halen in het omvangrijke corpus. Om die reden valt neosemie buiten het bestek van dit onderzoek.

Wedergeboorte. Sommige woorden kunnen het geluk hebben om na hun dood - of schijndood - wedergeboren te worden. Meestal wordt een woord weer tot leven gewekt voor komisch of ironisch effect.

Door de frequenties van woorden in die levensfasen in een staafdiagram te plotten kunnen we inzicht krijgen in hoe een woord evolueert. Verschillende woorden worden namelijk al vele jaren voor hun opkomst geattesteerd (Grieve et al. 2017). Door een frequentieverloop op te stellen kan men de fasen identificeren en eventueel relateren aan gebeurtenissen in de *community* waarin het neologisme gebruikt wordt.

Een methode om tot zo'n staafdiagram of grafiek met frequenties te komen is *Google Trends*. Dat is een interface die toont hoeveel een bepaald woord ingegeven is als zoekopdracht in Google. Onderzoek van Jiang (2021) ziet in enkele Chinese woorden hetzelfde patroon terugkomen: een snelle opkomst-afname patroon met vier

stadia. Tijdens het eerste stadium wordt het neologisme populair en neemt de frequentie toe. Daarna bereikt de populariteit haar piek voor een korte tijd. Afname begint in het derde stadium wanneer het woord minder populair en minder frequent wordt. Ten slotte is het woord niet meer populair en raakt het in onbruik (Jiang 2021). Ook ik zal in dit onderzoek eerst vertrekken van frequenties zoals Renouf (2013) en Jiang (2021) om een descriptief overzicht te bieden van de levenscyclus van enkele neologismen.

2.3. Lexicografisch perspectief

Naast het descriptieve perspectief, is er ook een lexicografische kijk op neologismen. Lexicografie houdt zich o.a. bezig met de vraag: met welke criteria moeten we rekening houden om een nieuw woord op te nemen in een woordenboek? Een woord moet voldoen aan inclusiecriteria en niet vallen onder de exclusiecriteria om opgenomen te kunnen worden in een woordenboek. Die criteria hangen natuurlijk af van de taal. Zo is er in de Romaanse lexicografie meer aandacht voor criteria voor neologismen dan in de Engelse lexicografie (Freixa 2020). Vakliteratuur (zie overzicht in Freixa 2020) deed al menig voorstel voor zulke criteria, waaronder morfologie, semantische transparantie, vroegste datering en frequentie. Aan die laatste twee wordt vaak de meeste aandacht geschonken.

Lexicografen moeten dus bepalen wat in hun taal een neologisme inhoudt. Er bestaat namelijk weinig tot geen consensus onder lexicografen van specifieke neologismenwoordenboeken over wat precies een "nieuw" woord inhoudt en hoe vroeg de vroegste datering moet zijn. In het *Algemeen Nederlands Woordenboek* (ANW) wordt een woord beschouwd als neologisme als het ontstaan is in of na 2000.¹ *Den Danske Ordbog* ("Het Deense Woordenboek") legt de grens dan weer 10 jaar eerder, in 1990. Trap-Jensen (2020) geeft wel toe dat dat arbitrair is en eerder omwille van praktische, project-interne redenen.

¹ Een uitzondering daarop vormen woorden die door neosemie een nieuwe betekenis hebben gekregen. Een woord kan dus al voor het jaar 2000 ontstaan zijn en later een neologisme worden met een nieuwe betekenis. Neosemie valt echter buiten het bestek van dit onderzoek, dat zich focust op formele neologismen. Dat is vooral om praktische redenen; de omvang van het gebruikte corpus (CHN) laat niet toe manueel te controleren welk woord welke betekenis heeft in welke context.

De bekendste criteria voor woordenboekopname van neologismen zijn de FUDGE-criteria van Metcalf. *FUDGE* staat voor "Frequency", "Unobtrusiveness", "Diversity of users and situations", "Generation of forms and meanings" en "Endurance of the concept" (Metcalf 2002). Hieronder wordt elk criterium kort uitgelegd.

Frequency. Hoe frequenter een nieuw woord gebruikt wordt, hoe groter de kans dat het zal overleven. Een precies getal kan daar niet opgeplakt worden, maar als een woord in meerdere contexten en meerdere media (zoals kranten, sociale media, gesproken taal...) voorkomt, is het al veelbelovend.

Unobtrusiveness. Een nieuw woord moet er normaal en onopvallend uitzien. Dat kenmerk hangt af van de taal waarin het woord gebruikt wordt. De morfologische en fonologische structuur verschilt immers van taal tot taal. Als een woord eruitziet zoals andere woorden in een taal, heeft het een grotere kans om opgenomen te worden in het lexicon.

Diversity of users and situations. Als een woord in diverse situaties en door een divers publiek gebruikt kan worden, heeft het een hogere overlevingskans. Als met andere woorden zowel jong als oud, m/v/x, hoogbegaafd, laagbegaafd... in verschillende situaties zoals op school, thuis of op het werk kunnen kiezen voor het woord, is de kans groot dat het langer zal leven.

Generation of forms and meanings. Ook de productiviteit van een neologisme speelt een rol, zoals al vermeld door Renouf (2013). Als een neologisme bijvoorbeeld via derivatie of samenstelling nieuwe woorden kan voortbrengen, is het vaak een succesvol woord. Zo heeft het neologisme *blog* al meerdere samenstellingen en derivaties voortgebracht, zoals *tuinblog*, *reisblog*, *blogger*, *bloggen*...

Endurance of the concept. Verder helpt het ook als het concept dat uitgedrukt wordt met het neologisme een blijvend concept is. Wetten, uitvindingen, woorden uitgevonden door bekende mensen... zijn goede kandidaten. Dat kan ook in verband worden gebracht met de term "hypostase" (in Schmid *hypostatization*). Het bestaan van een woord impliceert het bestaan van het concept dat het woord denoteert (Schmid 2008). Sommige woorden hypostaseren meer dan andere (cf. 2.4.), wat de levensloop ook kan beïnvloeden.

Een neologisme krijgt dan een FUDGE-score op 10 op basis van die kenmerken. Hoe hoger het scoort, hoe groter de kans dat het zal overleven en hoe groter de kans dat de lexicograaf besluit het woord op te nemen in het woordenboek.

Die score is echter niet altijd eenduidig of eenvoudig te achterhalen. Bovendien is het een arbeidsintensieve manier die subjectiviteit van de beoordelaar kan bevatten. Daarom is er in dit onderzoek voor gekozen om te werken met kenmerken die eenduidiger zijn en automatisch te achterhalen zijn. In een volgend hoofdstuk (3.4.) worden die kenmerken voorgesteld.

Zoals al aangehaald door Metcalf (2002) speelt frequentie dus een belangrijke rol in woordenboekopname. Traditioneel wordt neologismenonderzoek dan ook uitgevoerd op basis van frequenties. Hoe vaak komt een nieuw woord voor, op welke frequentie legt men de grens om een neologisme geëtableerd te verklaren en wanneer is een woord dood? Een woord dat slechts één keer voorkomt heeft minder kans om in het woordenboek terecht te komen dan neologismen die vaak voorkomen. Frequentie heeft zijn nut dan ook al bewezen op verschillende vlakken in kwantitatief onderzoek over lexicale verandering. Onderzoek naar grammaticalisatie en semasiologische verandering heeft baat bij frequenties omdat het kan helpen veranderingen te identificeren, beschrijven en uitleggen. Frequenties zijn ook nuttig bij onderzoek naar institutionalisering, lexicalisatie en onomasiologische verandering, aangezien men aan de hand van die frequenties de opkomst van nieuwe woorden en de competitie tussen synoniemen kan bijhouden (Grieve et al. 2017). Sommige woordenboeken, zoals het *Digitales Wörterbuch der deutschen Sprache* (DWDS), nemen zelfs een grafiek op bij elk woord met het gebruik op basis van frequentie in Duitse corpora (Kosem et al. 2021).

Deze thesis heeft ook een lexicografische invalshoek aangezien het het *Woordenboek van Nieuwe Woorden* gebruikt als input voor neologismen. Schat het WNW goed in of een neologisme al dan niet in het woordenboek moet komen? Gebruikt het met andere woorden goed voorspellende inclusie- en exclusiecriteria? Hierbij moet wel worden opgemerkt dat ik het voordeel heb dat ik kan terugblikken, terwijl een lexicograaf voorspellend te werk gaat. In hoofdstuk 3.1. wordt verder ingegaan op de rol van het WNW en *Neoloog* in deze studie.

2.4. Sociolinguïstisch en cognitief-linguïstisch perspectief

Enkele bedenkingen bij de frequentie benadering van neologismen in de lexicografie komen uit de sociolinguïstiek en de cognitieve linguïstiek. Binnen deze benadering zal ik twee tradities bespreken: de ene past sociolinguïstiek toe op lexicografisch onderzoek (Freixa), de andere is puur theoretisch-sociolinguïstisch van aard (Zenner, Würschinger & Schmid). Die twee scholen of tradities hebben dus een ander einddoel, maar beide kunnen dienen als inspiratie voor neologismenonderzoek.

Freixa (2020) argumenteert dat frequentie wel degelijk een belangrijke factor is in het bepalen of een neologisme opgenomen moet worden in het woordenboek of niet, maar het is niet per se dé bepalende voorwaarde. Daarom is het verschil tussen institutionalisering en woordenboekopname belangrijk. Institutionalisering is hoe vaak een gemeenschap een bepaald woord gebruikt. Zo kan het zijn dat een woord minder vaak voorkomt, omdat het fenomeen dat het aanduidt minder frequent is, maar dat het wel in het woordenboek is opgenomen. Het betekent echter ook niet dat als een woord geïnstitutionaliseerd is in een taal, dat het ook in het woordenboek staat. Zo wordt voor het concept "selfie" in het Spaans de Engelse benaming "selfie" wel gebruikt en is het dus geïnstitutionaliseerd, maar het is als "autofoto" opgenomen in het woordenboek (Freixa 2020). Institutionalisering en opname in het woordenboek zijn dus geen synoniemen van elkaar.

Het proces van institutionalisering kan nuttig zijn om woordenboekopname beter te begrijpen. Schmid (2008) beschrijft dat tussen het eerste voorkomen van een woord en de opname ervan in een taal er drie fenomenen plaatsvinden: "lexicalisatie", "institutionalisering" en "hypostase" (*hypostatization* in Schmid 2008). "Lexicalisatie" is "het opkomen van specifieke semantische, orthografische, fonologische of syntactische eigenschappen" (Schmid 2008) waardoor het een zelfstandig woord wordt. "Institutionalisering" is een sociopragmatisch proces waarbij het woord verspreid en gebruikt wordt door een gemeenschap van sprekers. "Hypostase" is een cognitief proces waarbij het concept geassocieerd wordt met het woord dat dat concept uitdrukt (Schmid 2008). Alle drie fasen moeten doorlopen worden om een woord te etableren en eventueel in het woordenboek op te nemen.

Frequentie is een relatief goede indicator van institutionalisering en woordenboekopname, maar er kan ook rekening gehouden worden met wat sprekers

zelf *dictionarizable* achten. Zij hebben namelijk intuïtief een beeld van de frequentie van een woord. Er is dan ook een verband tussen positieve evaluatie van een woord en de institutionalisering ervan (Freixa 2020). Ook weten sprekers goed welke soort woorden opgenomen worden in woordenboeken en welke niet.

Hiermee samenhangend is de attitude van sprekers ten opzichte van neologismen en woorden in het algemeen. In onderzoek van Klosa-Kückelhaus & Wolfer (2020) wordt de hypothese gesteld dat leenwoorden in het Duits trager worden aanvaard dan Duitse neologismen. Dat wordt onderzocht op basis van de frequentie van de neologismen. Uit het onderzoek blijkt echter weer dat frequentie geen goede voorspeller is van acceptatie. De frequentieverlopen tonen aan dat leenwoorden niet per se minder worden aanvaard dan Duitse neologismen. Frequentie kan namelijk beïnvloed zijn door verschillende factoren. Woorden uit dezelfde frequentieklasse kunnen toch een heel ander verloop vertonen (zie figuur 11). Ook hieruit blijkt dus dat de frequentie benadering van neologismen niet dé bepalende factor is voor het succes van neologismen.

Een alternatief voor frequentieverlopen dat licht kan werpen op de overleving van neologismen en dat komt uit de sociolinguïstiek is "diffusie". Würschinger (2021) onderzoekt bijvoorbeeld de verspreiding van Engelse neologismen op Twitter om de verschillen in hun diffusie in kaart te brengen. Hij vertrekt van frequenties, maar daaruit blijkt dat de frequentieverlopen de diffusiegraden met wisselend succes weten te ontdekken. Ze missen belangrijke informatie over de temporele dimensie van gebruiksprofielen van lexicale innovaties (Würschinger 2021). Daarom gebruikt hij nog een andere analysemethode, namelijk sociale netwerkanalyses. Zo blijkt dat sommige neologismen die volgens hun frequentie wijdverspreid lijken toch niet erg sociaal gediffuseerd zijn. Die neologismen zijn vaak erg veranderlijk of politiek getint (bv. *alt-left*) en komen dus ook slechts voor in *communities* met gelijkgezinde mensen. Würschinger (2021) besluit dat temporele en sociale informatie een belangrijk onderdeel zijn van de studie van lexicale innovatie en beter nuanceert dan frequentie(verlopen). Daarom wordt ook in dit onderzoek de diffusie van de neologismen in onderzocht. Het gaat hier niet om diffusie op Twitter, maar om diffusie tussen twee variëteiten van het Nederlands: Nederlands-Nederlands en Belgisch-Nederlands.

Nog een ander onderzoek dat sociolinguïstiek toepast op de studie van lexicale innovatie - meer bepaald leenwoorden - is dat van Zenner et al. (2012). Ook zij bespreken hoe andere methodes dan frequentieverlopen inzicht kunnen bieden in het succes van bepaalde woorden. In plaats van frequentie als maat van succes stellen zij een conceptgebaseerde onomasiologische aanpak voor. Daarmee willen ze de variatie in succes van Engelse leenwoorden die personen beschrijven (zoals *manager*) in het Nederlands onderzoeken. Een van de conclusies daaruit is dat hoe frequenter het concept is dat het leenwoord aanduidt, hoe minder succes het leenwoord zal hebben. Hoe frequenter een concept is, hoe meer *entrenched* het is in het lexicon, waardoor nieuwe alternatieven - leenwoorden - minder ingang zullen vinden. Dat is natuurlijk van toepassing op dit onderzoek, aangezien nieuwe woorden ook in competitie moeten gaan met het lexicon dat al langer bestaat. Daarom zal het verschil tussen leenwoorden en inheemse woorden op vlak van overleving worden onderzocht, alsook synonymie. Uit het onderzoek van Zenner et al. (2012) blijkt ook dat kortere leenwoorden (in syllaben) succesvoller zijn, wat aansluit bij de bevindingen van Van de Velde & Keersmaekers (2020) op Griekse woordenschat (cf. 2.5).

2.5. Evolutionair-linguïstisch perspectief

Een andere insteek komt uit de evolutionaire taalkunde. In tegenstelling tot de eerder vermelde takken van de taalkunde die neologismen vooral frequentieel behandelen, gebruikt de evolutionaire taalkunde andere statistische modellen. Die methodes komen uit de biologie en worden toegepast op levende wezens. Een voorbeeld daarvan is een survivalanalyse, zoals Kaplan-Meier en Cox Proportional Hazard modellen. Dat zijn testen die de overlevingskans van patiënten, of in dit geval woorden, kunnen berekenen.

In het onderzoek van Van de Velde & Keersmaekers (2020) wordt het Grieks van de vierde eeuw v.C. tot de achtste eeuw evolutionair onder de loep genomen. Via een survivalanalyse onderzoeken ze welke factoren de levensduur van woorden bepaalt. Uit de Kaplan-Meier curves blijkt dat woorden met een lage frequentie en die fonetisch langer zijn een kleinere overlevingskans hebben. Werkwoorden hebben een grotere kans op overleven dan adjectieven en substantieven.

Vergelijkbaar onderzoek (Van de Velde 2020) op Nederlandse woorden uit het *Woordenboek der Nederlandsche Taal* wijst ook uit dat frequente en korte woorden na 350 jaar een hogere kans op overleving hebben in vergelijking met zeldzame en lange woorden. Net zoals in het Grieks hebben werkwoorden een hogere overlevingskans dan adjectieven. Ook substantieven worden aan het succesvollere lijstje toegevoegd. Woorden die een hoge Levenshteinafstand hebben tot andere woorden uit het lexicon en er dus opvallend uitzien, hebben ook een hogere overlevingskans. Dat gaat in tegen het principe "Unobtrusiveness" van Metcalf (2002, cf. 2.3.), dat stelt dat woorden die lijken op andere woorden, en dus een lage Levenshteinafstand hebben, een hogere kans op overleving hebben.

In tegenstelling tot die twee onderzoeken die terug in de tijd gaan, bestaat er ook voorspellend evolutionair onderzoek. Geïnspireerd door biologische evolutie en de coronapandemie, trachten Jiang et al. (2021) te voorspellen wanneer de piek in populariteit van neologismen precies zal vallen. Daarvoor gebruiken ze een computationeel model uit de epidemiologie. Op basis van drie variabelen - de vatbare, geïnfecteerde en herstelde individuen die besmet worden door de host - wordt het mogelijk de piek te voorspellen. In taalkundig onderzoek is de host de persoon die het nieuwe woord tegenkomt. Die neemt het op in zijn/haar/hun lexicon en geeft het door aan vatbare individuen die dan besmet zijn. Als het besmette individu het woord niet wil gebruiken, is hij hersteld en is hij geen host meer in de levenscyclus van een neologisme. Wanneer het aantal herstelden hoger is dan het aantal geïnfecteerden start de afname van het virus, of in dit geval de populariteit van een woord. Het statistische model van Jiang (2021) kan dat inflectiepunt voorspellen.

Een punt van verschil dat hierbij op te merken valt is dat evolutionair onderzoek meestal op een macrodiachrone schaal wordt uitgevoerd. Het overspant dus meerdere eeuwen. Het gaat dan ook niet om neologismen, maar om woorden die al lange tijd opgenomen zijn in het lexicon. Deze thesis tracht echter op microdiachroon niveau over een tijdspanne van 18 jaar nieuwe woorden te analyseren.

2.6. Vraagstelling

Dit overzicht van bestaande studies rond neologismen maakt duidelijk dat de focus tot nu toe vooral ligt op onderzoek gebaseerd op frequentieverlopen. Bovendien nemen zulke studies slechts enkele woorden individueel onder de loep. Daarbij wordt ook weinig aandacht besteed aan de invloed van bepaalde kenmerken op het succes van neologismen en zij behouden vaak niet de kortlevende, "onsuccesvolle" neologismen. Een ander perspectief op diachrone veranderingen in het lexicon komt uit de recente studies van Van de Velde & Keersmaekers (2020). Zij gaan aan de hand van survivalanalyses na welke kenmerken zorgen voor een hogere overlevingskans. Zulke studies bevinden zich echter op macrodiachroon niveau en hebben geen betrekking op neologismen. Survivalanalyses kunnen echter een interessante meerwaarde bieden in neologismenonderzoek, aangezien ze rekening houden met de woorden die overlijden, in tegenstelling tot frequentieverlopen. Een geaggregeerd, microdiachroon overzicht van de levensloop en overlevingskans van neologismen en welke kenmerken voor succes zorgen is een lacune in het onderzoeksveld, zeker voor het Nederlands.

Het innovatieve karakter van dit onderzoek is dus tweeledig. Ten eerste werd in traditioneel neologismenonderzoek de survivalanalyse tot nu toe nog niet toegepast, maar werd gefocust op de frequentieverlopen van "succesvolle" lexicale innovaties waarbij kortlevende neologismen buiten beschouwing gelaten werden. Ten tweede werkt men in bestaand evolutionair onderzoek van het lexicon macrodiachroon, terwijl ik microdiachroon te werk zal gaan. Een bijkomende vraag daarbij is of patronen die op macrodiachrone schaal voorkwamen - met name het succes van frequente, unieke en korte woorden, en verba en nomina - ook op een microschaal van 18 jaar opduiken. Met andere woorden: welke kenmerken beïnvloeden de levensduur van neologismen? En, als afgeleide toepassing, kunnen survivalanalyses lexicografen helpen bij het opstellen van goed voorspellende inclusiecriteria voor neologismenwoordenboeken?

3. Methodologie

3.1. Data

De gebruikte data voor dit onderzoek komen uit verschillende hoeken. Enerzijds zijn er de bronnen die zorgen voor de meetpunten: neologismen. Die bronnen zijn een woordenboek (*Woordenboek van Nieuwe Woorden*) en een databank van kandidaat-neologismen (*Neoloog*) voor dat woordenboek. De detectie van neologismen zoals besproken in 2.1. heb ik dus overgelaten aan een al bestaand identificatiesysteem ontworpen door lexicografen. Zelf neologismen detecteren valt namelijk buiten het bestek van deze thesis. Anderzijds is er het corpus (*Corpus Hedendaags Nederlands*) waaruit de afhankelijke variabelen - de frequenties en dateringen van de neologismen - gehaald kunnen worden.

3.1.1. *Woordenboek van Nieuwe Woorden* (WNW)

Het *Woordenboek van Nieuwe Woorden* is een initiatief van het Instituut voor de Nederlandse Taal (INT) en is een online woordenboek met neologismen vanaf het jaar 2000. Sinds 2018 worden er nieuwe woorden in verzameld en beschreven: o.a. de betekenis, informatie over grammatica, spelling en etymologie worden vermeld. Ook de vroegste datering is opgenomen en de bron waarin het woord voor het eerst gebruikt is. De huidige (2021) versie van het WNW bevat 6802 woorden, behandeld in 2616 trefwoorden (INT WNW 2021).

Het woordenboek is corpusgebaseerd, met het ANW-corpus als basis. Dat is een corpus van meer dan 100 miljoen woorden dat voor het *Algemeen Nederlands Woordenboek* is samengesteld. Het bevat bronnen uit allerlei domeinen. Ook het *Corpus Hedendaags Nederlands* (cf. 3.2.) en het internet worden als bron gebruikt. Een bijkomende, onrechtstreekse en beperktere stroom van neologismen die potentieel in het WNW terecht komen komt uit *Neoloog* (cf. 3.1.2.), dat speciaal ontworpen is voor het WNW. Niet enkel blijvende neologismen worden in het WNW opgenomen, maar ook veel woorden die kortere tijd bestaan. 2628 woorden uit het WNW zijn ook al in het *Algemeen Nederlands Woordenboek* opgenomen (INT ANW

2021). Het WNW fungeert in deze studie niet enkel als dataset, maar ook als informatiebron voor de verklarende variabelen (cf. 3.4.).

3.1.2. *Neoloog*

Neoloog (2015) is een computertool en neologismendatabank ontwikkeld door en voor het INT door Jan Niestadt, Rob van Strien en Mathieu Fannee. Het wordt gebruikt door de lexicografen van het ANW en het WNW als hulp bij het selecteren van mogelijke lemma's. De tool is gebaseerd op het CHN, het *Corpus Hedendaags Nederlands* (cf. 3.2.). Nieuwe woorden - woorden die volgens de computer voor het eerst voorkomen - worden in Nederlandse en Vlaamse kranten en magazines in het CHN gedetecteerd en opgeslagen in een lijst. Daar staat ook het jaartal en de maand bij. Daarna moeten lexicografen manueel checken of elk kandidaat-neologisme effectief een neologisme is (Waszink 2020). In *Neoloog* kunnen zij bij elk neologisme met behulp van een filter "neo" aanduiden of het er een is ("true"), of dat het er geen is ("false"). Bij de kandidaat-neologismen die dan effectief neologismen zijn volgens de lexicografen is er ook nog een filter "ANW" om aan te duiden of het woord in het WNW opgenomen zal worden.² Bovenop de neologismen uit het WNW heb ik dus ook naar de neologismen uit *Neoloog* gekeken waarbij de filter "neo" op "true" stond.

Een belangrijke opmerking is dat aangezien *Neoloog* pas sinds 2015 neologismen verzamelt, het vroegste voorkomen van veel neologismen gedateerd wordt in 2015, terwijl ze eigenlijk al vroeger voorkomen. Om een meer betrouwbare datering te bekomen, heb ik ervoor gekozen met het originele corpus te werken waaruit *Neoloog* de neologismen haalt, namelijk het *Corpus Hedendaags Nederlands*.

3.2. Afhankelijke variabelen: *Corpus Hedendaags Nederlands*

De afhankelijke variabelen in dit onderzoek zijn de frequentieverlopen en de informatie over vroegste en laatste datering van de neologismen uit het WNW en *Neoloog* in het

² *Neoloog* claimt dat alle neologismen waar de filter "ANW" op "true" staat ook uiteindelijk opgenomen zijn in het WNW (voordien heette het WNW immers ook ANW). Wanneer ik echter via een *inner merge* checkte hoeveel woorden uit *Neoloog* met "true" bij de filter "ANW" ook effectief in het WNW waren opgenomen, bleken dat er relatief weinig te zijn. Van de 1300 zogenaamde opnames, stonden er slechts 158, of 12%, in het WNW.

Corpus Hedendaags Nederlands (CHN). Net zoals het WNW en *Neoloog* is ook het CHN ontwikkeld door het INT in 2014 en dient het als input voor het WNW en *Neoloog*. Het corpus bevat Nederlands tekstmateriaal waarvoor het INT toestemming heeft gekregen om het intern te gebruiken, waaronder kranten als *NRC* en *De Standaard*. Vier variëteiten van het Nederlands zijn vertegenwoordigd: Nederlands-Nederlands, Belgisch-Nederlands, Nederlands van Suriname en van de Nederlandse Antillen. De data omvatten ook de corpora van het eerdere INL, exclusief het corpus juridisch Nederlands. In totaal zijn er 2.568.010 documenten in opgenomen, goed voor 1.000.627.401 tokens (12/03/2021 16:57:56).

De periode die vertegenwoordigd wordt in het corpus begint al in 1962, maar pas vanaf 2003 is er voldoende materiaal voor het Nederlands-Nederlands en het Belgisch-Nederlands, wat het studieobject van dit onderzoek is. Het corpus bestaat uit kranten, boeken, blogs, chats, magazines, neergeschreven gesproken taal... (INT CHN 2021). Enkel kranten tussen 2003 en 2021 werden gebruikt voor dit onderzoek, omdat daar de grootste continue en eenduidig gedateerde datastroom voor beschikbaar was.

Wanneer de neologismen uit het WNW dan in het CHN opgezocht werden, bleek de vroegste datering niet altijd overeen te komen. Dat kan te wijten zijn aan neosemie, homonymie... met woorden in het CHN of doordat de vroegste datering in het WNW fout is. Er zit dus een bias in de neologismen in het WNW, maar de informatieve aard van het WNW weegt daartegen op. Enorm veel eigenschappen die manueel veel extra werk zouden vragen staan in het WNW.

Ook het CHN is niet vrij van bias. Door schrijffouten in de teksten en decoderings- en parseringsfouten in het corpus zijn niet alle zoekopdrachten en resultaten volledig betrouwbaar, in het bijzonder de resultaten van de zoekopdrachten om productiviteit te meten (cf. 3.4.2.).

3.3. Dataverwerking en -selectie

De lijst van neologismen uit het WNW bestaande uit 2616 woorden moest eerst nog wat opgeschoond en aangevuld worden om deze studie mogelijk te maken. Ten eerste

heb ik 26 woorden die het gevolg zijn van neosemie en polygenese niet behouden,³ respectievelijke voorbeelden zijn *biopas* en *bakfietsen*. Dat was zowel uit theoretische als praktische redenen. Neosemie en polygenese vallen - zoals al eerder vermeld - buiten het bestek van deze studie en het is door het grote aantal data in het CHN moeilijk om te achterhalen welk woord in welke betekenis gebruikt wordt als het meerdere betekenissen heeft. Ten tweede heb ik 46 homonieme woorden, wat vaak afkortingen zijn, niet behouden omwille van dezelfde praktische redenen. De 158 dubbels die zowel in Neoloog als het WNW stonden werden ontdebeld. Daarna werden de kenmerken (cf. 3.4.) van de overgehouden neologismen die in het WNW staan via een pythonscript⁴ uit de XML van het WNW gehaald en weggeschreven naar een bestand. Voor het kenmerk "Domein" heb ik enkele domeinen samengenomen, omdat sommige domeinen in het WNW eigenlijk subdomeinen van elkaar zijn, zoals "flora en fauna" van "natuur en milieu" (voor een exact overzicht cf. 3.4.4.). De kenmerken voor de neologismen uit *Neoloog* werden manueel toegevoegd (cf. 3.4.).

In het CHN bestaan er verschillende zoekfuncties. Je kunt filteren op lemma, woord, woordsoort... Ook het medium, het jaar, de maand, de taalvariëteit... kunnen geselecteerd worden om zo een subcorpus te bekomen. De precieze filters die voor dit onderzoek gebruikt werden zijn: "woord" als zoekfilter, "newspaper" als medium en taalvariëteit afhankelijk van de zoekopdracht al dan niet opgesplitst in Nederlands-Nederlands en Belgisch-Nederlands. Daarna werden de resultaten gegroepeerd per jaar (voor het frequentieverloop) of op lemma (voor de productiviteit van het linker- of rechterlid). Via een API-request in Python werd zo alle informatie uit het CHN weggeschreven naar hetzelfde bestand waarin de kenmerken staan. Het CHN werd dus in eerste instantie gebruikt om de frequenties met de jaartallen bij te bekomen, waardoor de leeftijd van elk neologisme kon berekend worden.

Wanneer een woord uit het WNW niet voorkwam in het CHN wil dat niet zeggen dat het woord niet bestaat, maar dat het uit een andere bron komt en dat die bron niet opgenomen is in het CHN. Omdat er voor die woorden dan geen data bestaat, heb ik die ook niet behouden. Zoals al eerder vermeld loopt de onderzochte periode van 2000-2021, gebaseerd op de grens van het WNW (Waszink 2016). In het CHN is er

³ Voor een lijst van de behouden en niet behouden woorden, zie Appendices I & II.

⁴ Voor de pythonscripts, zie Appendix III.

echter pas voldoende materiaal voor beide taalvariëteiten vanaf 2003. Als een woord dus stierf voor 2003 volgens het CHN, werd het ook niet behouden. Zo kwam de dataset van het WNW uit op 1511 woorden, of 57% van het totale WNW. De dataset van neologismen uit *Neoloog* bestaat uit 2293 woorden, of 73% van de neologismen uit 2015 (N=3121). De verwijderde woorden uit *Neoloog* (N=828, 27%) kwamen al voor 2000 voor in het CHN, of waren dubbels, en werden om die reden verwijderd uit de dataset.

3.4. Onafhankelijke variabelen

Er zijn verschillende kenmerken die de levenscyclus van een neologisme kunnen beïnvloeden. Die kenmerken dienen in dit onderzoek als onafhankelijke of verklarende variabelen. Ik zal aan de hand van die variabelen proberen te verklaren wat het succes van een neologisme bepaalt. In dit hoofdstuk worden enkele kenmerken en wat de literatuur daarover zegt voorgesteld. Voor elke onafhankelijke variabele wordt ook de operationalisering en de verdeling van het kenmerk in de dataset gegeven. Net zoals in de literatuurstudie is de artificiële opdeling in fonetische, morfologische, sociolinguïstische of cognitief-linguïstische en semantische kenmerken gebeurd met het oog op overzichtelijkheid.

3.4.1. Fonetische kenmerken

Aantal karakters

Een eerste fonetisch kenmerk dat het succes van een neologisme kan bepalen is woordlengte. Uit onderzoek van Van de Velde (2020) op woorden uit het *Woordenboek der Nederlandsche Taal* (WNT) blijkt dat korte woorden een grotere overlevingskans hebben dan lange woorden (cf. 2.5.). Hetzelfde geldt voor woorden uit het (Oud-)Grieks (Van de Velde & Keersmaekers 2020, cf. 2.5.). Ook uit Zenner et al. (2012, cf. 2.4.) blijkt dat kortere leenwoorden succesvoller zijn. Hoewel die studies geen betrekking hebben op neologismen, is het plausibel dat ook korte neologismen langer leven dan lange. Het aantal karakters van de neologismen werd aan de dataset toegevoegd met behulp van Excel. Daarna werden de neologismen onderverdeeld in "kort", "medium" en "lang". Korte neologismen bevatten tussen de 3 en 8 karakters,

medium tussen 9 en 14, en lange tussen 15 en 32 karakters. Die opdeling resulteerde in 451 korte, 1980 medium en 1373 lange woorden.

Levenshteinafstand

Niet alleen de lengte van het woord zelf, maar ook de afstand tot andere woorden kan zorgen voor een hogere overlevingskans. De Levenshteinafstand berekent het aantal inserties, deleties en substituties dat er moet gebeuren om het ene woord te transformeren in het andere. Zo is de Levenshteinafstand tussen *bofkip* en *plofkip* 2. Het onderzoek van Van de Velde (2020) wijst uit dat woorden die een grote gewogen Levenshteinafstand hebben en er dus uniek uitzien, langer blijven leven (cf. 2.5.). Dat gaat in tegen Metcalfs (2002) criterium "Unobtrusiveness", dat redeneert dat nieuwe woorden onopvallend moeten zijn om te overleven. Zoals al eerder gezegd heeft Van de Veldes (2020) onderzoek geen betrekking op neologismen, dus dit onderzoek zal moeten uitwijzen welke hypothese het haalt bij nieuwe woorden.

Om de Levenshteinafstand te berekenen werd er gebruik gemaakt van de Levenshteinimplementatie in Python van Keuleers (OLD20 2010). De lijst van neologismen kan zo worden vergeleken met een lijst genaamd SUBTLEX-NL (Keuleers et al. 2010). Dat is een database van Nederlandse woordfrequenties gebaseerd op 44 miljoen woorden uit ondertitels. Het gaat om een psycholinguïstische dataset, wat dus niet helemaal aansluit bij de aard van deze thesis. De evolutionaire taalkunde vertrekt echter wel vaak van psycholinguïstisch geïnspireerde verklaringen, waardoor de keuze voor SUBTLEX-NL wel gerechtvaardigd is. Het aangepaste Pythonscript vergelijkt dan de neologismen met de woorden uit SUBTLEX-NL, en slaat het woord met de laagste Levenshteinafstand en met de hoogste frequentie op.

De verkregen Levenshteinafstanden werden daarna nog zelf genormaliseerd of gewogen door de afstand te delen door het aantal karakters. Dat resulteerde in 193 neologismen met een lage gewogen Levenshteinafstand (afstand = 0-0,29), 1786 met een medium (afstand = 0,3-0,49) en 1825 met een hoge (afstand = 0,5-1)) gewogen afstand. De lage Levenshteinafstanden (0-0,29) werden dan nog onderverdeeld op basis van de frequentie van hun dichtstbijzijnde woord, namelijk 12 hoogfrequente (frequentie 5211-188.542), 59 mediumfrequente (frequentie 600-4587) en 122 laagfrequente (frequentie 158-591) woorden. De hypothese is dan dat erg lijken op

een hoogfrequent woord voor verwarring en dus een lagere overlevingskans zorgt (Van de Velde 2020).

Een opmerking hierbij is dat er geen rekening gehouden is met derivatieve morfologie bij het berekenen van de Levenshteinafstand. Zo zullen meervouden of afleidingen zorgen voor een heel kleine Levenshteinafstand. Die afleidingen zullen het oorspronkelijke woord niet in de weg zitten, maar net de interpretatie vergemakkelijken. Afleidingen zullen niet voor begripsverwarring of ongewenste synonymie zorgen (zie het principe van Humboldt 2011). Bovendien zijn niet alle woorden met dezelfde Levenshteinafstand even verschillend. Het verschil tussen *tinderen-kinderen* is groot genoeg om geen verwarring te zaaien. De "t" en de "k" hebben namelijk veel minimale paren. Andere lettercombinaties, zoals de "s" en de "z", zullen misschien weer wel verwarring veroorzaken. In het geval van *tinderen* zorgt de uitgang "-inderen" ook nog eens voor een gemakkelijkere herkenning, aangezien "-inderen" een productieve uitgang in het Nederlands is (bv. *hinderen, zinderen...*). Er zijn dus wel meer gevallen waarin woorden op elkaar lijken, maar niet per se verwarring teweegbrengen. In deze studie is daar geen rekening mee gehouden, het is dus stof voor vervolgonderzoek.

3.4.2. Morfologische kenmerken

Lidwoord

Uit onderzoek (Blom et al. 2008; Hulk & Cornips 2006) blijkt dat het lidwoord *het* steeds minder gebruikt wordt en misschien ooit zal verdwijnen. Daar bestaan verschillende verklaringen voor. Blom et al. (2008) stellen vast dat zowel L1-kinderen, als Marokkaanse L2-leerders van het Nederlands (jong en oud) het lidwoord *de* overgeneraliseren in het nadeel van het lidwoord *het*. Als verklaring daarvoor geven ze het gebrek aan houvast: het lidwoord in het Nederlands is niet erg transparant. Bovendien wordt *het* niet alleen als lidwoord gebruikt, maar ook als onbepaald en persoonlijk voornaamwoord (zoals in "Het regent"). Kinderen gebruiken *het* als onbepaald voornaamwoord zelfs sneller correct dan als lidwoord (Hulk & Cornips 2006). Als verklaring daarvoor wordt gegeven dat het lidwoord *het* nu eenmaal minder vaak voorkomt. In het *Corpus Gesproken Nederlands*, dat bestaat uit 6 miljoen woorden uit mondeling taalgebruik, komt *het* in slechts 1 op de 5 gevallen als lidwoord

voor. Het is dus plausibel dat het lidwoord *het* kan zorgen voor een lagere overlevingskans bij neologismen. De lidwoorden van de neologismen in het WNW konden uit de XML van het woordenboek gehaald worden, in *Neoloog* ontbrak die informatie. De mogelijke lidwoordaanduidingen in het WNW zijn "de" (N=1041), "het" (N=249), "de of het" (N=43) en "geen" (N=178).

Geslacht

Samenhangend met het lidwoord, is er het geslacht van een woord. Voor *het*-woorden is het eenvoudig: zij zijn onzijdig. *De*-woorden daarentegen kunnen mannelijk of vrouwelijk zijn. Door de eeuwen heen heeft het Standaardnederlands zich ontwikkeld van een drie-generasysteem naar een twee-generasysteem, waardoor mannelijk en vrouwelijk samen zijn gevallen en het gendersysteem niet zo transparant meer is. Tussen dialecten zijn er wel nog gradatieverschillen: in West-Vlaamse dialecten is het verschil tussen mannelijk en vrouwelijk minder zichtbaar dan in bijvoorbeeld Brabantse dialecten, "door het ontbreken van een apart masculien onbepaald lidwoord (in casu *ne(n)*)" (Cornips & De Vogelaer 2009). Ook uit onderzoek (De Vogelaer 2006) naar het gebruik van pronomina bij kinderen blijkt de hersemantisering van het lidwoord. Zodra het concept waarnaar ze verwijzen iets abstracts is, doen ze dat met het voornaamwoord *het*, ook al is het een *de*-woord. Daardoor wordt het lidwoord *het* steeds minder gebruikt als lidwoord. Het gendersysteem van het Nederlands staat dus door meerdere factoren onder druk en de tijd zal moeten uitwijzen welke veranderingen nog zullen plaatsvinden. Voor dit onderzoek is een mogelijke hypothese dat door de hersemantisering van het lidwoord *het* de onzijdige woorden minder succesvol zijn dan *de*-woorden, of mannelijke en vrouwelijke woorden. De verwachtingen zijn dat de curve voor "geslacht" dus gelijkaardig aan die van lidwoord zal zijn.

De geslachten van de neologismen uit het WNW konden via een pythonscript uit de XML van het WNW gehaald worden. Informatie over het geslacht van de neologismen uit *Neoloog* was niet beschikbaar. De mogelijke geslachtsaanduidingen in het WNW zijn: "mannelijk"; "vrouwelijk"; "onzijdig"; "mannelijk of vrouwelijk"; "mannelijk of onzijdig"; "vrouwelijk of onzijdig"; "mannelijk, vrouwelijk of onzijdig"; "niet van toepassing". Deze werden in Excel samengevoegd tot "mannelijk" (N=537),

"vrouwelijk" (N=244), "onzijdig" (N=254), "onzeker" (N=304) en "niet van toepassing" (N=172).

Woordsoort

Ook zijn er bepaalde woordsoorten die een langer leven beschoren zijn dan andere. Uit het onderzoek van Van de Velde (2020) blijkt dat nomina en verba hogere overlevingskansen hebben dan adjectieven (zie sectie 2.5.). Uit Van de Velde & Keersmaekers (2020) volgt dan ook nog dat verba een hogere overlevingskans dan nomina hebben. Adjectieven verdwijnen dus sneller uit de woordenschat. Een mogelijke reden daarvoor is dat adjectieven niet-noodzakelijke zinsdeelstukken zijn. Het fenomeen kan ook verklaard worden aan de hand van hypostase. Substantieven objectiveren het concept dat ze denoteren en hypostaseren meer dan adjectieven (Schmid 2008). Je kunt je met andere woorden bij een substantief sneller iets inbeelden dan bij een adjectief. De mogelijkheid om een woord cognitief te conceptualiseren is dus een belangrijk kenmerk voor neologismen die willen overleven. De woordsoort van de neologismen uit het WNW kon via een pythonscript uit de XML van het WNW gehaald worden. De woordsoort van de neologismen uit *Neoloog* werden manueel toegevoegd. De mogelijke woordsoorten zijn: "substantief" (N=3560), "adjectief" (N=74), "werkwoord" (N=138) en "overige" (uitroep, tussenwerpsel...) (N=32).

Woordvorming

Neologismen kunnen op verschillende wijzen morfologisch in elkaar zitten. Zelden zijn neologismen totaal arbitrair uitgevonden. Het Nederlands leent zich uitstekend tot het vormen van samenstellingen en afleidingen, zonder daarvoor spaties of koppeltkens nodig te hebben, zoals het Engels. Bovendien zijn samenstellingen makkelijker te leren en onthouden. Als je een nieuw woord tegenkomt is het handig als je de samenstellende delen al kent (Waszink 2016). De woordvorming van de neologismen uit het WNW werd via een pythonscript uit de XML van het WNW gehaald. De woordvorming van de neologismen uit *Neoloog* werden manueel toegevoegd. Wegens de scheve verhouding van samenstelling in het WNW (N=1136) tegenover andere woordvormingen, zoals "afleiding" of "blend" (bv. *boerkin*), (N=375) werd

besloten om woordvorming een binair kenmerk te maken: "samenstelling" (N=3372) of "geen samenstelling" (N=432).

Bij de samenstellingen kan de productiviteit van het linker- en/of rechterlid ook een rol spelen bij het succes van een neologisme, zoals volgt uit Renoufs levensfase "productiviteit" (2013, cf. 2.2.) en Mecalfs (2002) criterium "generation of forms and meanings" (cf. 2.3.). Hij heeft het wel over de productiviteit van het volledige neologisme en niet de samenstellende delen ervan. De productiviteit werd berekend met een API-request aan het CHN. De zoekopdracht voor het linkerlid was het linkerlid gevolgd door 4 of meer tekens. Zo werden vals positieven zoals afleidingen eindigend op "-er", "-aar" of "-ing" vermeden. Voor neologismen die als linkerlid een werkwoord hadden, werd handmatig het linkerlid omgezet naar de stam om dan de stam gevolgd door 4 of meer tekens als zoekopdracht te gebruiken. Verder mochten de hits ook niet het lemma zelf zijn. Voor het rechterlid was de zoekopdracht het rechterlid voorafgegaan door 1 of meer tekens. Ook hierbij mochten de hits niet het lemma zelf zijn.

Bij woorden met een relatief lang linker- of rechterlid (meer dan 4 tekens) is die werkwijze redelijk betrouwbaar. Woorden met een korter linker- of rechterlid daarentegen lopen het risico veel vals positieven te geven. Een voorbeeld daarvan is *gen* in de betekenis van "een stukje DNA". *Gen* is zo kort dat *leugen*, *dagen*, *wegen*... ook als hits worden geïnterpreteerd. Gelukkig waren zulke gevallen redelijk beperkt in de dataset.

Zo wordt dus het aantal samenstellingen met het linker- of rechterlid van het neologisme uit het CHN gehaald. Die frequenties werden dan opgedeeld in lage (frequentie 0-736, N=585), medium (frequentie 752-2997, N=574) of hoge (frequentie 3014-234.886, N=352) productiviteit. De productiviteit kon enkel voor de neologismen uit het WNW berekend worden, omdat die samenstellingen al gesplitst waren.

3.4.3. Sociolinguïstische en cognitief-linguïstische kenmerken

Diffusie

Geïnspireerd door het onderzoek van Würschinger (2021, cf. 2.4.) zal ook de invloed van diffusie op de overlevingskans van neologismen onderzocht worden. De diffusie van een woord kan meerdere vormen aannemen. Het kan verspreid worden over

verschillende domeinen, bv. *tsunami* dat eerst enkel een natuurkundige term was met als betekenis "extreem hoge golf uit de zee". Daarna werd tsunami in andere domeinen gebruikt om "een overvloed aan iets bepaalds" weer te geven. Dat type diffusie is niet zuiver sociolinguïstisch, het combineert semantische verschuiving (metafoor) met sociale verspreiding (van natuurkundigen naar de bredere bevolking). Daardoor wil ik de artificiële opsplitsing in fonetische, morfologische, sociolinguïstische en semantische kenmerken nog eens benadrukken; er is een overlap tussen die perspectieven. Een tweede type van diffusie is de verspreiding in verschillende *communities*. Het woord *bestie* kwam eerst voor in (online) jongerentaal, maar sinds het woord verkozen werd tot tienerwoord van 2021 (in Vlaanderen) heeft het ook ingang gevonden bij de iets oudere bevolking. Ook dit voorbeeld is niet zuiver sociolinguïstisch van aard, er is namelijk ook een semantische verschuiving. Waar *bestie* eerst naar jonge mensen verwijst, kan het nu ook naar oudere mensen verwijzen. Een laatste voorbeeld van diffusie is de verspreiding tussen taalvariëteiten. Dat soort van diffusie zal in dit onderzoek bekeken worden aan de hand van de twee grootste variëteiten van het Nederlands: Nederlands-Nederlands (NN) en Belgisch-Nederlands (BN). Komt het woord eerst voor in het NN en wordt het daarna gediffuseerd naar het BN, of omgekeerd? Komt het enkel in een van de twee variëteiten voor, of onmiddellijk in beide? De operationalisering van dit kenmerk verloopt via het CHN: via de API werd de variëteit van het eerste en laatste voorkomen opgehaald. Daarna werd in Excel toegevoegd of het neologisme "eerst in het BN en dan in het NN" voorkwam (N=360), of "eerst in het NN dan in het BN" (N=786); of "onmiddellijk in beide" (N=337); of "enkel in het BN" (N=1113); of "enkel in het NN" (N=1208). Daaruit blijkt overigens dat het *Woordenboek van Nieuwe Woorden* zich vooral richt op het Nederlands-Nederlands.

Herkomst

Van der Sijs (2002) verdeelt neologismen in vijf groepen op basis van hun herkomst: leenwoorden, afgeleiden van leenwoorden, inheemse woorden, letter(greep)woorden en woorden met onbekende herkomst. Neologismen die leenwoorden zijn hebben meestal als brontaal het Engels. Het is plausibel dat als men het de moeite vindt om een woord te lenen uit een andere taal, waar het woord al langer bestaat, het woord ook in het Nederlands een grotere overlevingskans heeft (cf. 2.4., Zenner et al. 2012).

Informatie over de herkomst van de neologismen in het WNW werd via een pythonscript uit de XML van het WNW gehaald. De herkomst van de neologismen ontbrak in *Neoloog*. Wegens de grote verscheidenheid in brontalen werd ervoor gekozen om herkomst een binair kenmerk te maken: "leenwoord" (N=411) of "inheems woord" (N=1100).

3.4.4. Semantische kenmerken

Domein

In de meeste woordenboeken worden woorden onderverdeeld in domeinen. Uit onderzoek van Trap-Jensen (2020) blijkt dat de meest productieve domeinen bij Deense neologismen computergebruik, sport, eten, communicatie en business zijn. Ook veel neologismen in het Nederlands vallen binnen het domein computergebruik (Waszink 2016). Voor het kenmerk "domein" werden enkele domeinen uit het WNW samengenomen omwille van praktische redenen. De samengevoegde domeinen zijn: "taal en literatuur" bij "kunst en cultuur"; "geschiedenis" bij "kunst en cultuur"; "techniek en technologie" bij "onderwijs en wetenschap"; "natuurkunde en scheikunde" bij "onderwijs en wetenschap"; "flora en fauna" bij "natuur en milieu"; "agricultuur en visserij" bij "natuur en milieu"; "feestdagen, herdenkingen, festiviteiten en evenementen..." bij "toerisme en recreatie". In *Neoloog* waren er geen domeinen aangeduid.

Gebaseerd op Waszink (2016) en Trap-Jensen (2020) is mijn hypothese dat woorden uit het domein "automatisatie, informatie en communicatie" (N=205) weleens een hogere overlevingskans zouden kunnen hebben dan de rest (N=847). Ook woorden die in meerdere domeinen (N=275) ondergebracht kunnen worden zijn misschien een langer leven beschoren dan die die in één domein voorkomen (N=777) omdat ze dan voldoen aan Metcalfs (2002) principe "diversity of users and situations". Domein hangt dus ook samen met diffusie (cf. 3.4.3.).

Synonymie

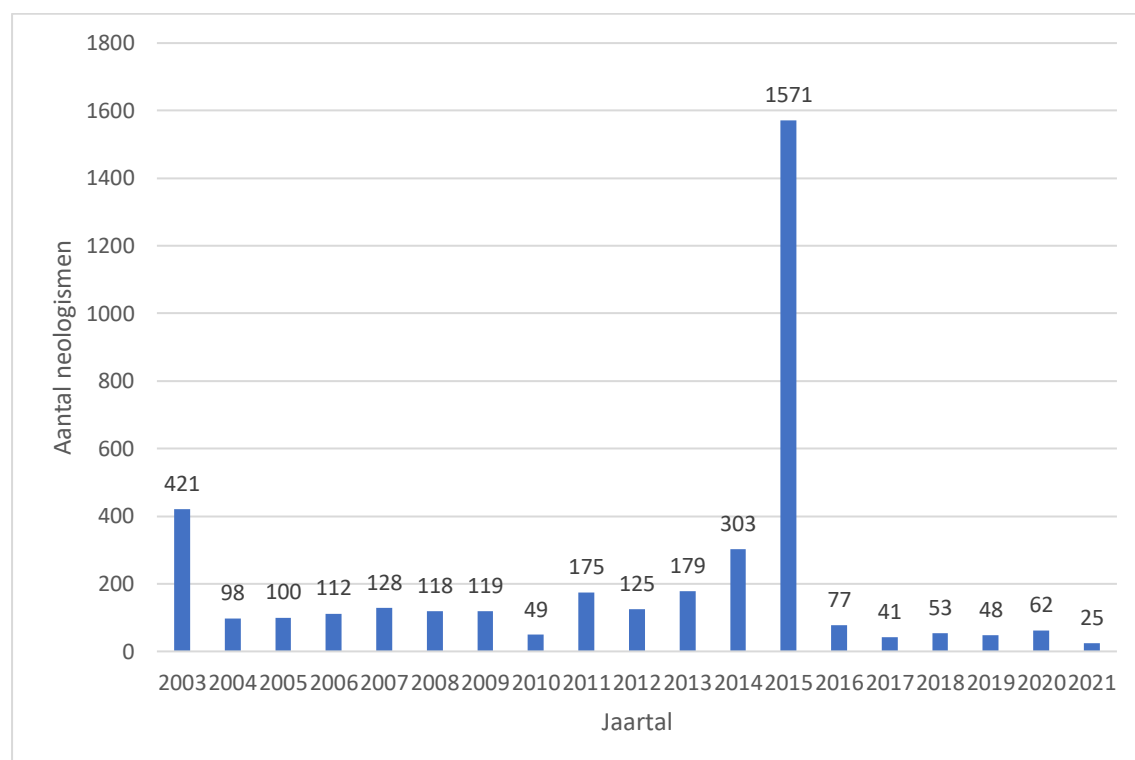
Een ander semantisch kenmerk dat een rol kan spelen in het succes van een neologisme is synonymie (zie sectie 2.4., Zenner et al. 2012). Wanneer een nieuw woord in competitie moet gaan met een ander woord dat hetzelfde betekent en dat al langer in het lexicon zit, kan dat nadelig zijn voor het neologisme. Tenzij het nieuwe

woord het concept efficiënter aanduidt, is het plausibeler dat het woord dat al langer bestaat de voorkeur krijgt van de taalgebruiker. In het WNW wordt bij enkele neologismen een aantal synoniemen opgelijst. Via een pythonscript kon in de XML van het woordenboek geteld worden hoeveel synoniemen er gekend zijn voor bepaalde neologismen. 715 neologismen hadden minstens 1 synoniem, voor de andere 796 was er geen bekend of was er tenminste geen opgenomen in het WNW. In *Neoloog* was er geen informatie beschikbaar over synonymie.

4. Beschrijvende analyse van enkele frequentieverlopen

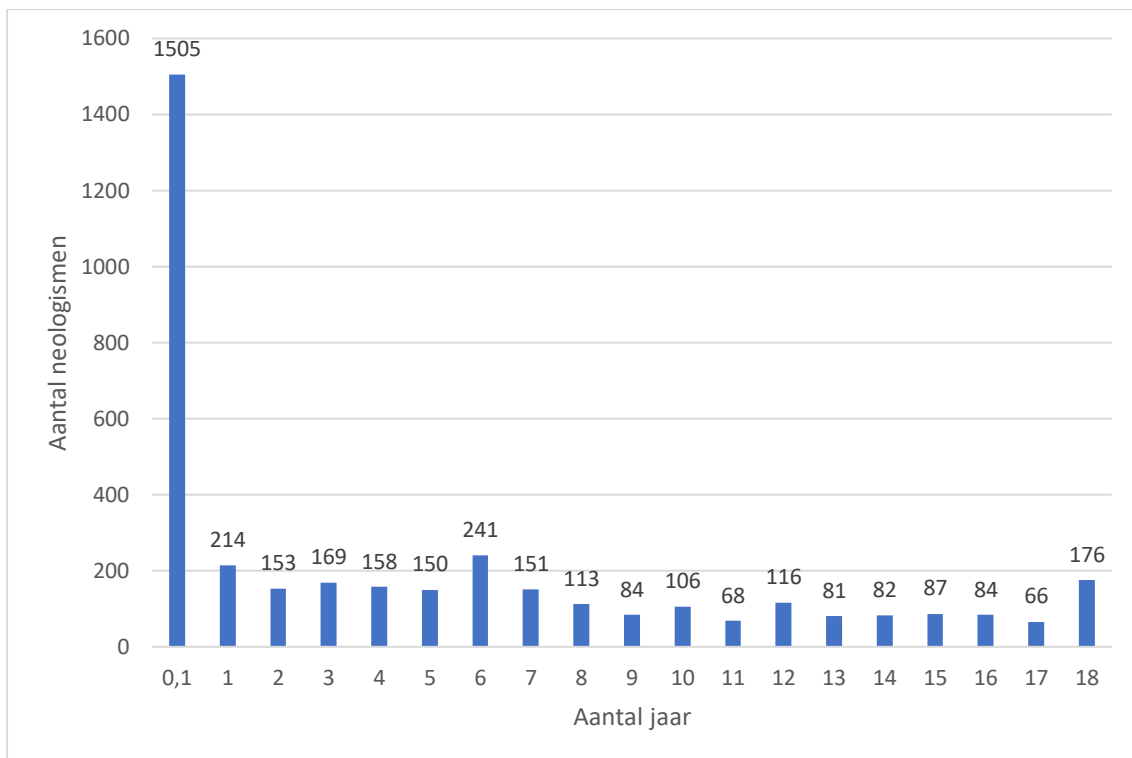
In wat volgt geef ik een beschrijvend overzicht van de dataset op basis van enkele staafdiagrammen en frequentieverlopen. Zoals al eerder vermeld is dat de traditionele benadering van neologismen. In dit onderzoek is een beschrijving op basis van frequentieverlopen echter niet de hoofdbedoeling, maar het kan wel inzicht bieden in de samenstelling van de dataset.

Als eerste volgt een overzicht van hoeveel neologismen in elk jaar van 2003 tot 2021 in deze studie opgenomen zijn. Zo krijgen we een zicht op hoeveel neologismen de kans hadden om bijvoorbeeld 18 jaar te worden, maar het misschien niet geworden zijn. Zoals al eerder vermeld zitten er ook neologismen die geboren zijn tussen 2000 en 2002 in de studie. Zij zijn in dit staafdiagram bij de neologismen uit 2003 geteld (2000: 174; 2001: 114; 2002: 93). Uit Figuur 1 blijkt dat uit 2015 het grootste aantal neologismen is opgenomen, wat vooral neologismen uit *Neoloog* zijn. Ook zijn er 421 woorden uit 2000-2003 die 18 jaar oud hadden kunnen worden, maar zoals verderop zal blijken zijn er 176, of 41%, daarvan effectief 18 jaar geworden (cf. Figuur 2).



Figuur 1: Aantal neologismen dat opgenomen is per jaar (N=3804).

Een indeling van de dataset van het WNW en *Neoloog* samen op basis van leeftijd⁵ maakt duidelijk dat het leeuwendeel van de neologismen geen lang leven beschoren is. Meer dan 39,6% van de neologismen wordt niet ouder dan een jaar. Meer dan de helft van die neologismen komt dan ook slechts één keer voor in het CHN. Het gaat dan vooral om neologismen die een kortstondige gebeurtenis, ontdekking... aanduiden, het gevolg zijn van de creativiteit van de auteur of misschien vaker in gesproken taal voorkomen.

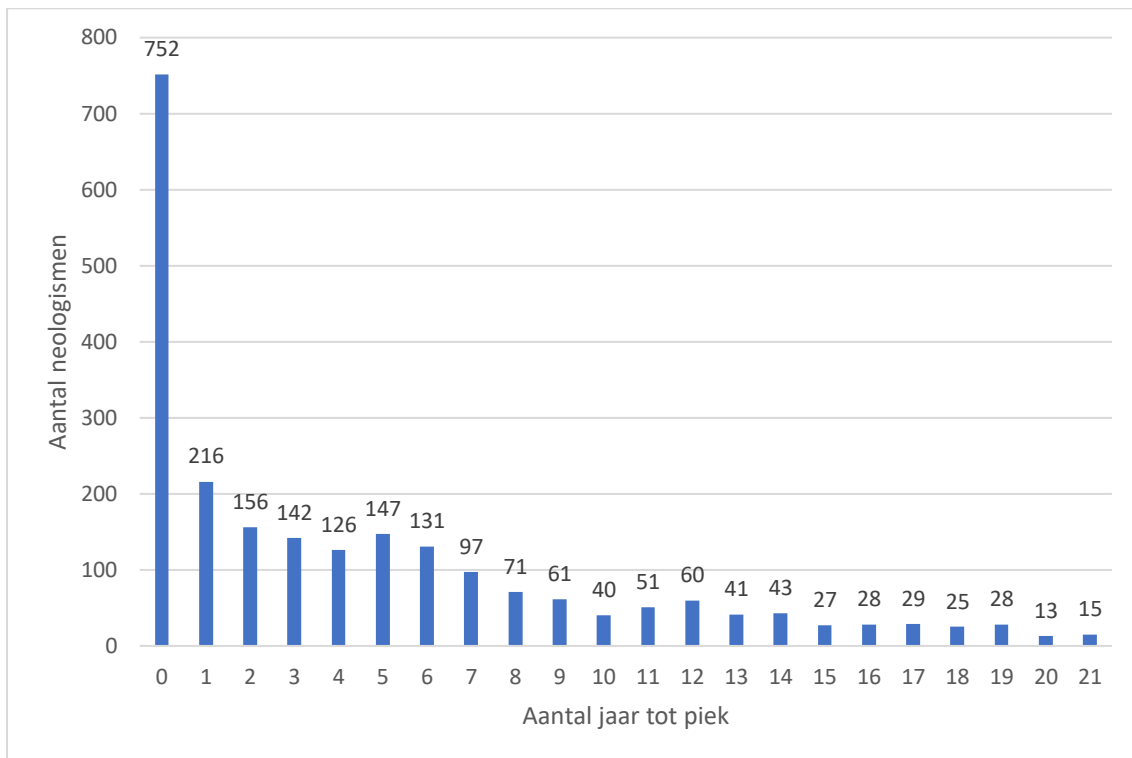


Figuur 2: Overlevingsduur van alle neologismen (N=3804), waarbij 0,1 staat voor minder dan 1 jaar oud.

Ook wanneer men kijkt naar hoeveel jaar na het ontstaan van het neologisme het zijn frequentiepiek bereikt, is de verdeling zeer scheef. Bijna 60% bereikt zijn piek al in zijn eerste levensjaar (0,1 staat voor minder dan 1 jaar oud). Zoals al eerder gezegd komt meer dan de helft daarvan slechts één keer voor en is de frequentie van die piek dus 1. Op de volgende figuur (Figuur 3) zijn de 1505 neologismen die geen jaar worden eruit gehaald omdat hun piek sowieso in het eerste jaar valt. Uit Figuur 3 blijkt dat 752

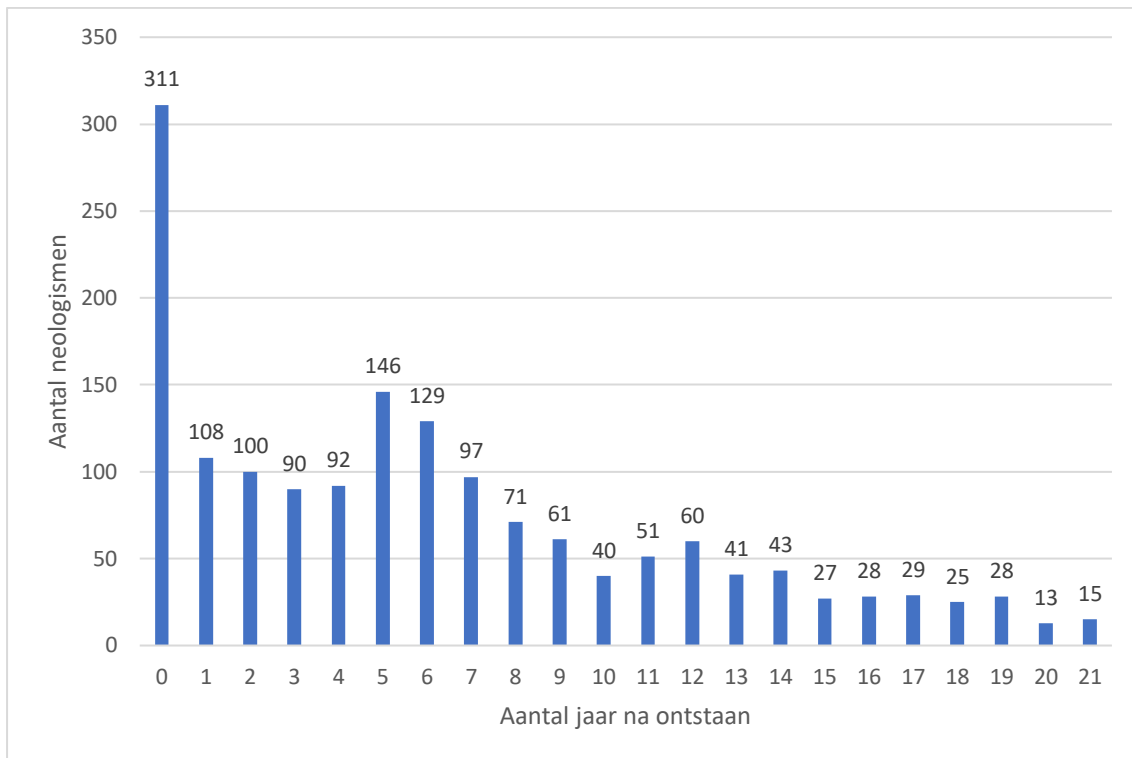
⁵ Zoals al eerder vermeld wordt hier pas geteld na 2003, omdat vanaf dan het grootste aantal data beschikbaar was in het CHN.

neologismen (die dus ouder dan 1 jaar worden) in hun eerste levensjaar hun piek al bereiken.



Figuur 3: Aantal jaar tot neologismen hun piek bereiken (N=2299).

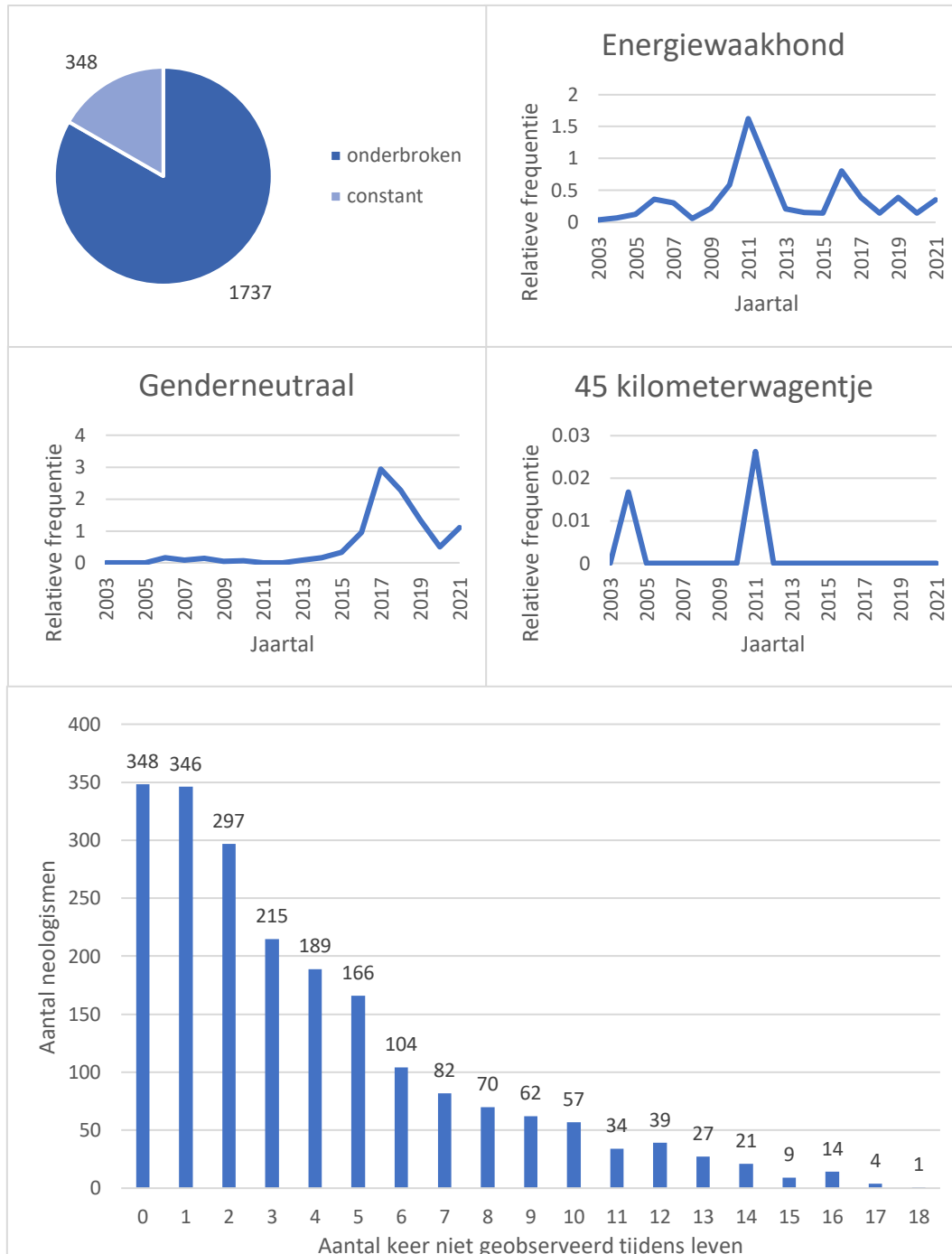
Om te beoordelen hoelang het duurt voor neologismen tot hun “volle wasdom” komen, is het minder relevant om te kijken naar diegenen die maar een kort leven beschoren zijn. Daarom kan het handig zijn om enkel naar de neologismen te kijken die minimum 5 jaar oud geworden zijn. Ook daaruit blijkt dat het merendeel (19%) van de neologismen zijn piek bereikt in zijn eerste levensjaar. In vergelijking met alle neologismen en de neologismen die ouder dan 1 worden is dat aantal wel kleiner.



Figuur 4: Hoeveel jaar na hun ontstaan bereiken neologismen van minstens 5 jaar oud (N=1605) hun piek?

Verder is het relevant om te weten hoe constant het neologisme geobserveerd wordt tijdens zijn leven. Als een neologisme als vroegste datering 2003 heeft en als laatste datering 2010, maar daartussen 0 keer geobserveerd wordt, kan men dan nog spreken van een levend neologisme? Om een overzicht van dat probleem te bieden, werd er geteld hoeveel neologismen minstens 1 keer tijdens hun leven niet geobserveerd werden. De neologismen die tussen 0 en 2 jaar oud werden, werden uit de telling gelaten omdat zij sowieso heel hun leven lang geobserveerd zijn. Van de 2085 neologismen waren er 1737 neologismen die minstens in een jaar niet voorkwamen tijdens hun leven. Verder werd er ook voor elk neologisme berekend hoeveel keer het tijdens zijn leven niet geobserveerd werd. Uit dat staafdiagram blijkt dat het aantal neologismen afneemt naarmate de discontinuïteit toeneemt. Een aantekening daarbij is wel dat er in het staafdiagram geen rekening is gehouden met de leeftijd van het neologisme. Als een neologisme dat 4 jaar leeft 3 keer niet geobserveerd is, is dat discontinuër dan een neologisme van 12 jaar dat 3 keer niet is geobserveerd. Er bestaan dus verschillende graden van continuïteit bij neologismen. Hieronder volgen enkele voorbeelden van woorden die doorheen hun leven constant

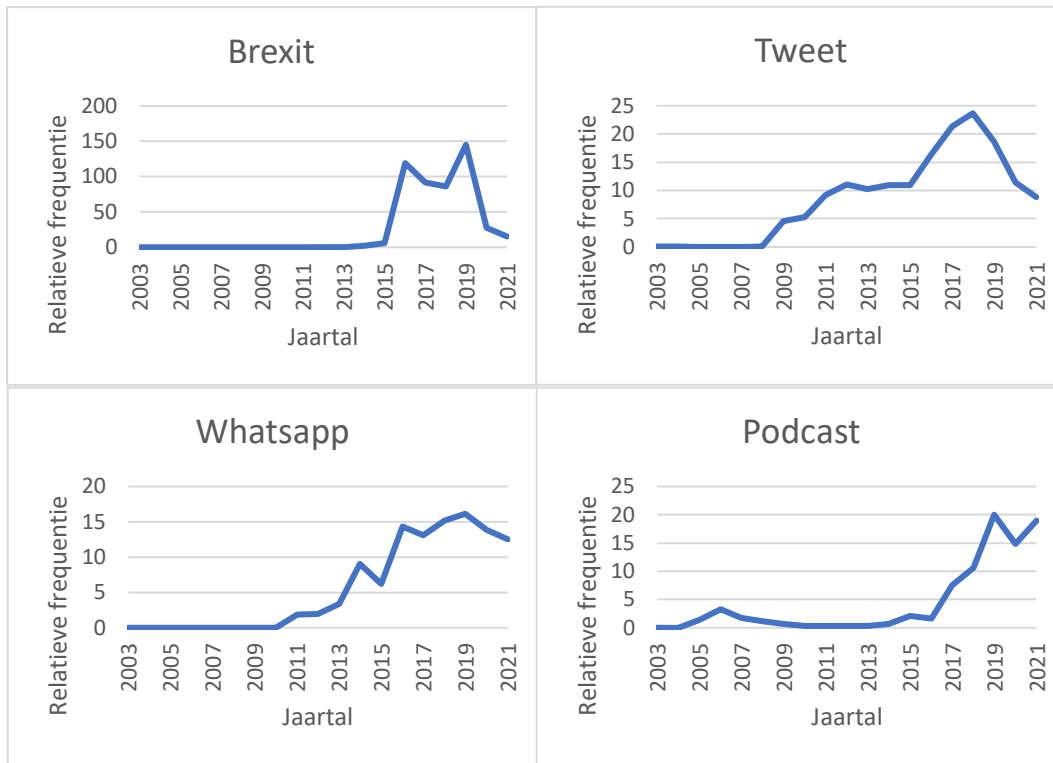
geobserveerd worden (*energiewaakhond*), enkele keren niet geobserveerd worden (*genderneutraal*) en enkel aan het begin en einde (*45 kilometerwagentje*). Er wordt gewerkt met relatieve frequenties, d.w.z. aantal voorkomens per miljoen woorden.



Figuur 5: Continuïteit van observatie (N=2085) & Verschillende graden van continuïteit.

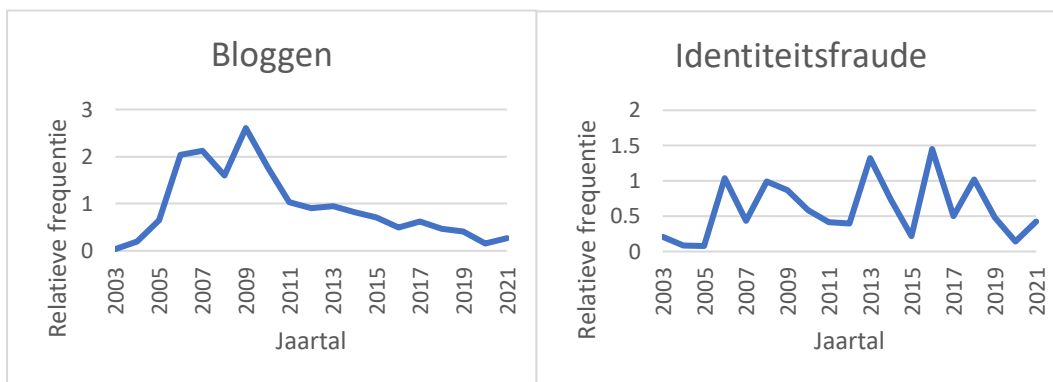
De dood van een neologisme is dus een problematisch gegeven. Daarom is in deze studie voor een simpele oplossing gekozen: als een woord na een bepaald jaar nooit meer opduikt, is het dood, ongeacht hoe vaak het tussen zijn ontstaan en dood voorkomt in het corpus. Het betrekken van de discontinuïteit in de survivalanalyses, eventueel als tijdsvariërende covariant, laat ik dus over aan vervolgonderzoek. Voor onderzoek op basis van frequentieverlopen is die discontinuïteit een groter probleem dan voor survivalanalyses. Enerzijds is het weinig interessant als er geen verloop is tussen ontstaan en dood, wat geldt voor kortlevende neologismen. Voor survivalanalyses is het wel relevant als er "kindersterfte" optreedt bij bijna de helft van de neologismen. Anderzijds zijn er neologismen die een discontinue frequentieloop hebben, waardoor die frequentieverlopen niet zo informatief zijn. Survivalanalyses kijken echter niet naar de frequenties tussen geboorte en dood, maar wel naar de overlevingsgraad. Voor deze thesis die zich focust op neologismen die vaak slechts een kort of discontinu leven beschoren zijn, zijn survivalanalyses dus informatiever dan frequentieverlopen.

Toch kan het interessant zijn om naar de frequentieverlopen van enkele individuele woorden te kijken om de verscheidenheid aan levenslopen duidelijk te maken. De vier frequentste woorden in de dataset zijn *Brexit*, *tweet*, *Whatsapp* en *podcast*. Opvallend is dat de vier frequentieverlopen een stijgend verloop kennen en het leenwoorden zijn binnen het domein van "automatisering, informatie en communicatie". Ze leven ook nog in 2021, wat al inzicht biedt in de overlevingskans van leenwoorden (cf. 5.1.3.).



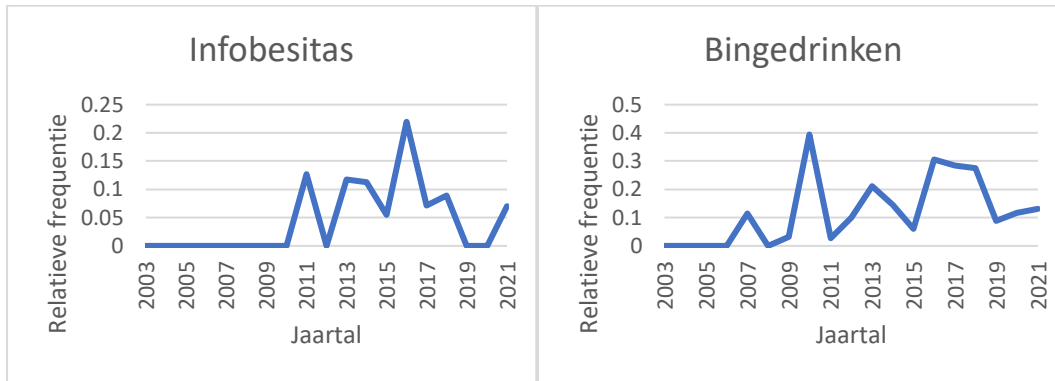
Figuur 6: Hoogfrequente neologismen.

De middelfrequente woorden kennen een grotere verscheidenheid aan frequentieverlopen. Bij *bloggen* daalt het frequentieverloop sterk na 2009, wat het verlies aan populariteit van blogs ten gevolge van de komst van web 3.0 reflecteert. In het frequentieverloop van *identiteitsfraude* duikt dan weer *seasonality* op met pieken en dalen die zich ongeveer om de twee jaar herhalen.



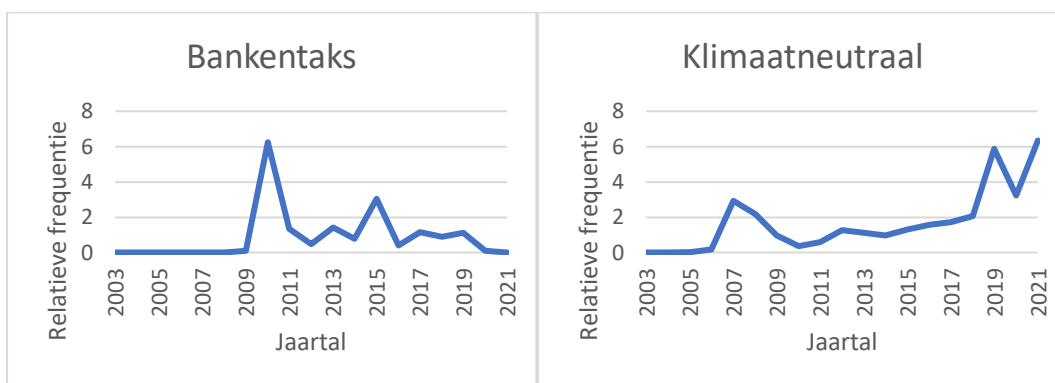
Figuur 7: Middelfrequente neologismen.

Ook laagfrequente woorden kennen eerder onregelmatige frequentieverlopen, zoals voor *infobesitas*. Toch kan er ook een stijgende trend gevonden worden in frequentieverlopen van laagfrequente woorden, zoals *bingedrinken*.



Figuur 8: Laagfrequente neologismen.

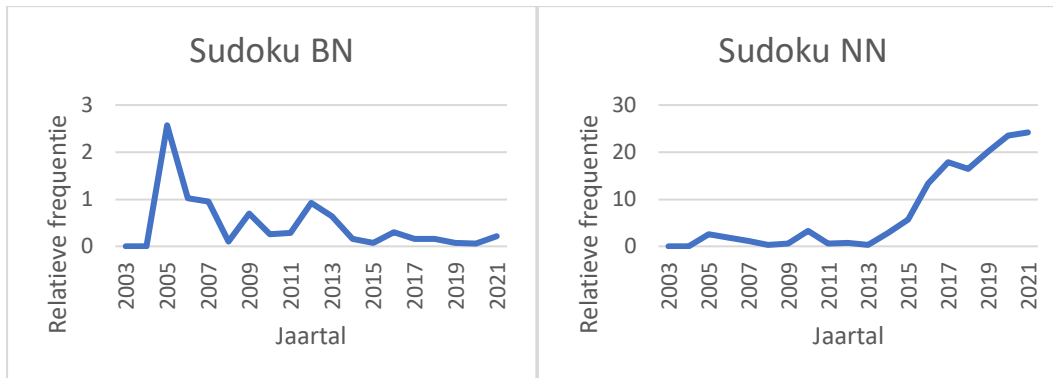
Zoals al eerder vermeld is er ook grote verscheidenheid in de frequentiepieken. Sommige woorden pieken vroeg, zoals *bankentaks*. Ook hier reflecteert deze piek iets uit de actualiteit: de economische crisis in de banksector in 2008. Het omgekeerde komt ook voor: een woord dat pas piekt 21 jaar na haar ontstaan, zoals *klimaatneutraal*. Het woord kwam al in 2003 voor, maar pas de laatste jaren door klimaatrampen, -marsen en -activisten is de aandacht voor klimaatneutraliteit gestegen, met als gevolg dat het woord ook vaker opduikt.



Figuur 9: Verscheidenheid in frequentiepieken.

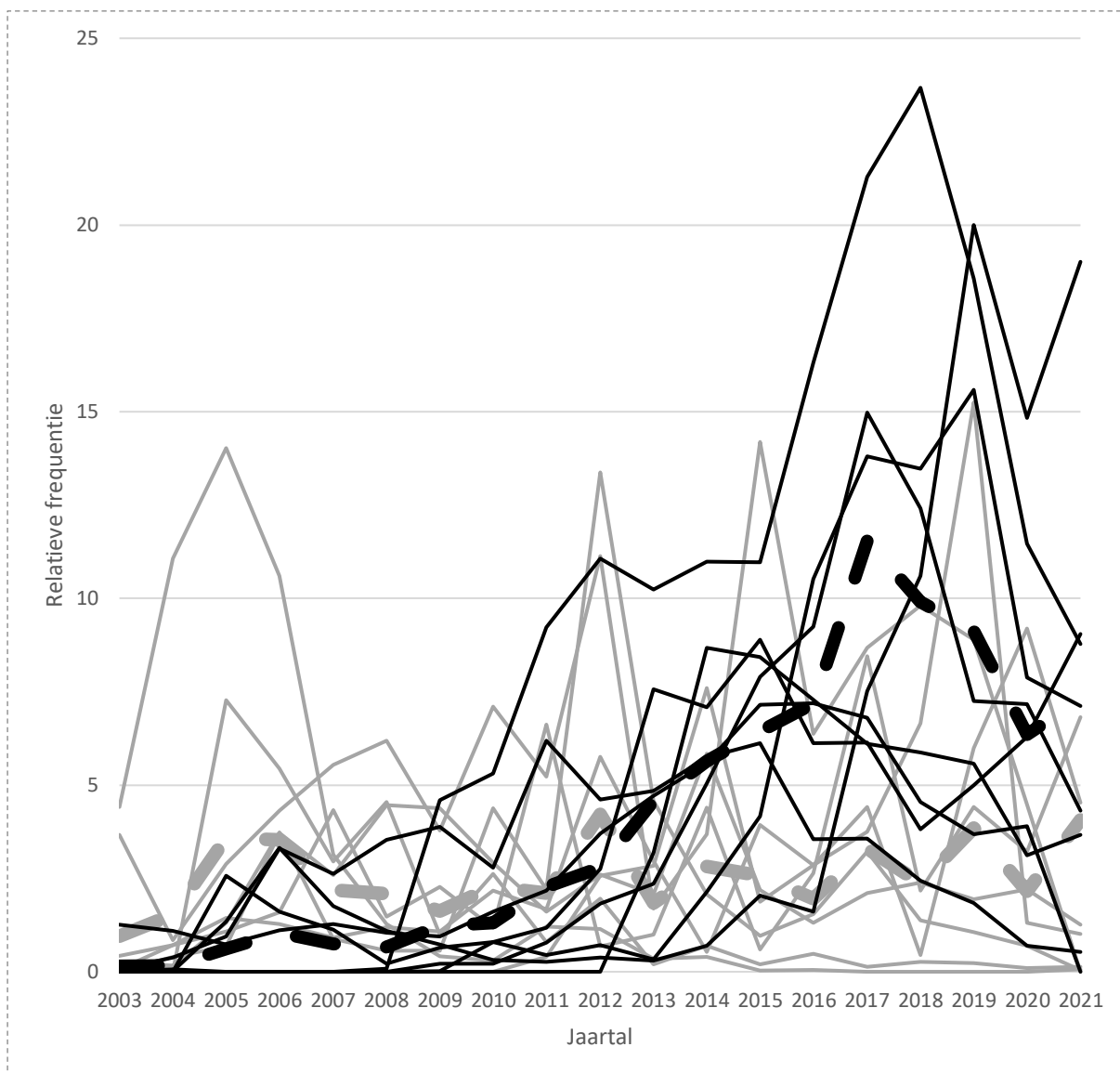
Interessant is ook een vergelijking tussen frequentieverlopen van eenzelfde neologisme in Belgisch-Nederlands en Nederlands-Nederlands. Voor de meeste neologismen zijn die frequentieverlopen sterk gelijkend, al is er hier en daar een

verschil. Een van de frequentste woorden in de dataset, *sudoku*, kent in het BN een piek in het begin van zijn leven, terwijl die piek in het NN pas veel later komt. Bovendien wordt het woord *sudoku* veel vaker in het NN gebruikt. Dat zou echter wel kunnen liggen aan de kranten die in het CHN zijn opgenomen. Misschien bieden Nederlandse kranten meer sudoku's aan dan Belgische?



Figuur 10: Verschil in frequentieverloop tussen BN en NN.

Geïnspireerd op het onderzoek van Klosa-Kückelhaus & Wolfer (2020) heb ik ook een frequentieverloop gemaakt van 9 frequente leenwoorden (zwart) versus 10 frequente inheemse neologismen (grijs). De inheemse woorden zijn *coronapas*, *generatiepact*, *kernuitstap*, *kinderpardon*, *kringwinkel*, *leefloon*, *levensloopregeling*, *zorgautoriteit*, *zorgpremie* en *zorgtoeslag*. De leenwoorden zijn *bitcoin*, *blogger*, *crowdfunding*, *hashtag*, *podcast*, *selfie*, *sudoku*, *tweet* en *wifi*.



Figuur 11: Frequentieverloop van 9 frequente leenwoorden (zwart) vs. 10 frequente inheemse neologismen (grijs). De stippellijnen zijn de gemiddelde relatieve frequentie.

Daaruit wordt duidelijk dat er niet onmiddellijk een patroon valt te ontdekken in de frequentieverlopen van leenwoorden versus inheemse woorden, behalve dan dat ze beide in relatieve frequentie stijgen tegen 2021. De gemiddelde relatieve frequentie van leenwoorden (stippellijnen) ligt ook hoger dan die van inheemse woorden (11,5 vs. 3,5 in 2017). Bovendien blijft de gemiddelde relatieve frequentie van inheemse woorden relatief constant, in tegenstelling tot de leenwoorden die pas in 2017 hun piek bereiken. Een mogelijke verklaring daarvoor is dat leenwoorden eerst een tijdje nodig hebben voor ze ingeburgerd geraken, tegenover inheemse woorden die constant in het lexicon blijven. De grafiek bewijst ook dat er een grote verscheidenheid aan

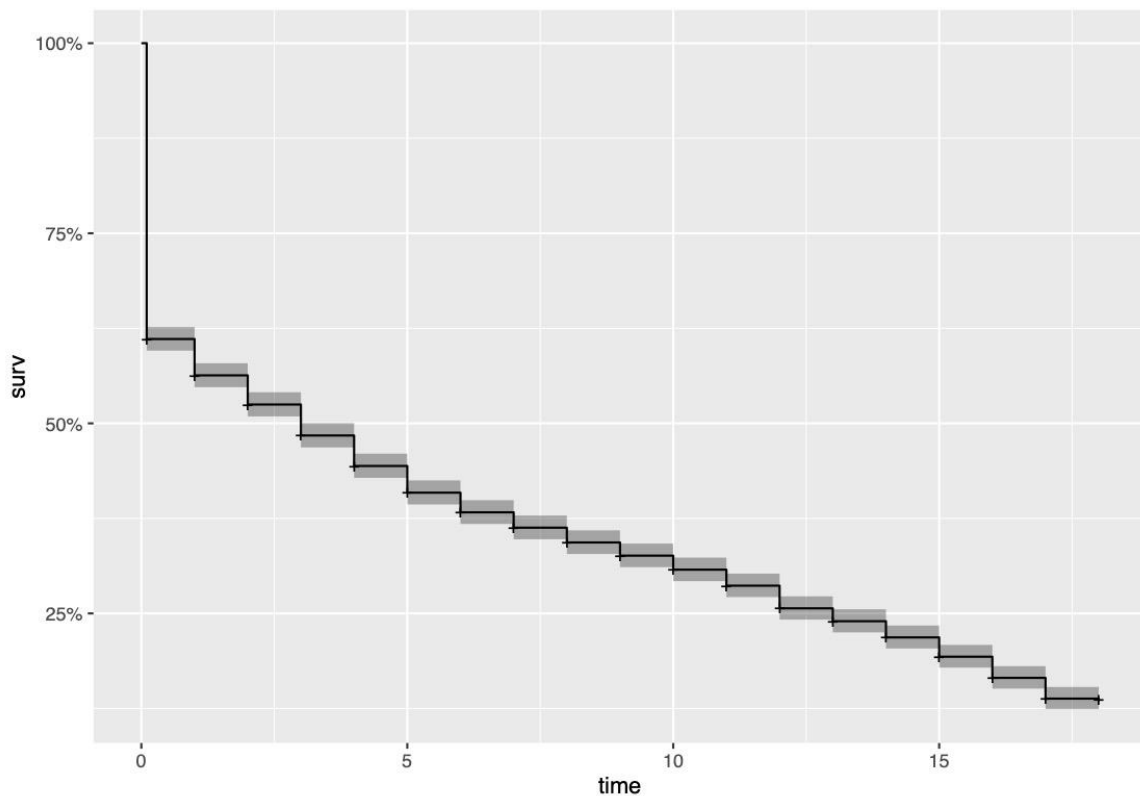
frequentieverlopen bestaat. Om de verschillen in overlevingskans tussen twee of meer variabelen te onderzoeken zijn frequentieverlopen dus niet de ideale methode. In het volgende hoofdstuk wordt de gekozen werkwijze van deze thesis - survivalanalyses - uitgelegd en verdedigd.

5. Resultaten van de survivalanalyses

5.1. Kaplan-Meier

Een Kaplan-Meier analyse is een bivariate survivalanalyse met een niet-parametrische benadering waarbij de tijd wordt opgedeeld in intervallen. Zo'n analyse wordt vaak toegepast in de biologie of geneeskunde om het succes van bijvoorbeeld geneesmiddelen te onderzoeken. De tijdsintervallen kunnen gaan van jaren tot dagen en zelfs minuten. Elk tijdstip waarop een patiënt of woord iets overkomt - in dit geval sterft - wordt het aantal doden gedeeld door het aantal overlevenden. Zo wordt de cumulatieve proportie van overlevenden op alle tijdstippen berekend en samengebracht in een survivaltabel en -curve (van Geloven 2018). Toegepast op de taalkunde is de frequentie van het woord dus minder van belang. In plaats daarvan wordt er gekeken naar de leeftijd van een woord en of het nog leeft in een bepaald jaar. Het resultaat is dan een percentage van hoeveel woorden met een bepaald kenmerk er nog leven na x-aantal jaar. Via de Log rank test kunnen ook de p-waardes berekend worden. Die kwantitatieve methode kan goed omgaan met de temporele aard van diachrone linguïstische data (Van de Velde & Keersmaekers 2020). De Kaplan-Meier analyses werden uitgevoerd in het programma *R* (R Core Team 2020) met behulp van de packages *survival* (Therneau 2022), *ggplot2* (Wickham 2016), *ggfortify* (Horikoshi & Tang 2018) en *survminer* (Kassambara et al. 2019).

Een Kaplan-Meier analyse op alle data (N=3804) zonder verklarende variabele wijst uit dat ongeveer 38,9% sterft al voor het 1 jaar oud is.⁶ Van de 421 neologismen (cf. Figuur 1) die 18 jaar oud hadden kunnen worden, zijn er 176, of 41%, effectief zo oud geworden. Een opmerking daarbij is dat we niet voor alle 3804 neologismen data hebben voor 18 jaar. Het voordeel aan een Kaplan-Meier analyse is dat die voor zo'n dataset toelaat neologismen in de survivalanalyse te betrekken die niet de volledige observeringstijd (18 jaar in dit geval) aanwezig zijn. De Kaplan-Meier vergelijkt namelijk telkens per jaar het aantal risicolopende neologismen met het aantal sterfgevallen. Als er dan neologismen wegvallen omdat ze overleven, worden die niet langer als risicolopend beschouwd.

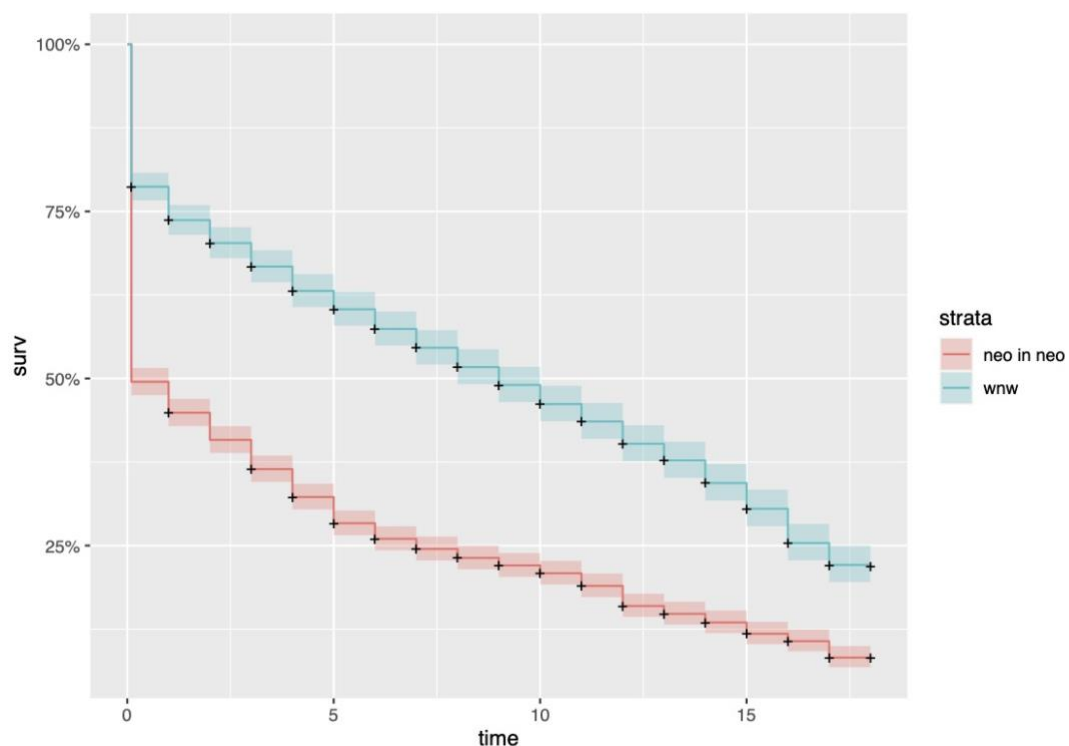


Figuur 12: Algemene survivalcurve (N=3804).

Wanneer we de dataset opsplitsen op basis van bron, dus in neologismen uit het WNW (N=1511) en neologismen uit *Neoloog* (N=2393), blijkt uit de vergelijking van de curves dat de neologismen die in het WNW staan een hogere overlevingskans hebben (Chisq= 345 on 1 degrees of freedom, $p = <0.001$) dan die uit *Neoloog*. Bovendien sterft er een kleiner aantal woorden (21,3%) uit het WNW in zijn eerste levensjaar in

⁶ Voor de survivaltabellen, p-waardes, chi-squares en pairwise comparisons, zie Appendix IV.

vergelijking met de massale "kindersterfte" (50,5%) die plaatsvindt in *Neoloog*. Dat bewijst dat de lexicografen van het WNW relatief goed voorspellende inclusiecriteria hebben gehanteerd om te bepalen welke neologismen in het WNW terechtkomen. De neologismen uit *Neoloog* hebben een lagere overlevingskans omdat de hoofdbedoeling van *Neoloog* natuurlijk detectie is en *Neoloog* niet meteen bepaalt welk woord in het woordenboek terecht komt. Aan de andere kant zijn het wel de lexicografen en annoteerders die manueel in *Neoloog* hebben aangeduid dat het neologismen zijn. Nogmaals merk ik hier op dat ik het voordeel heb in het verleden te gaan kijken, terwijl zij de toekomst probeerden te voorspellen.

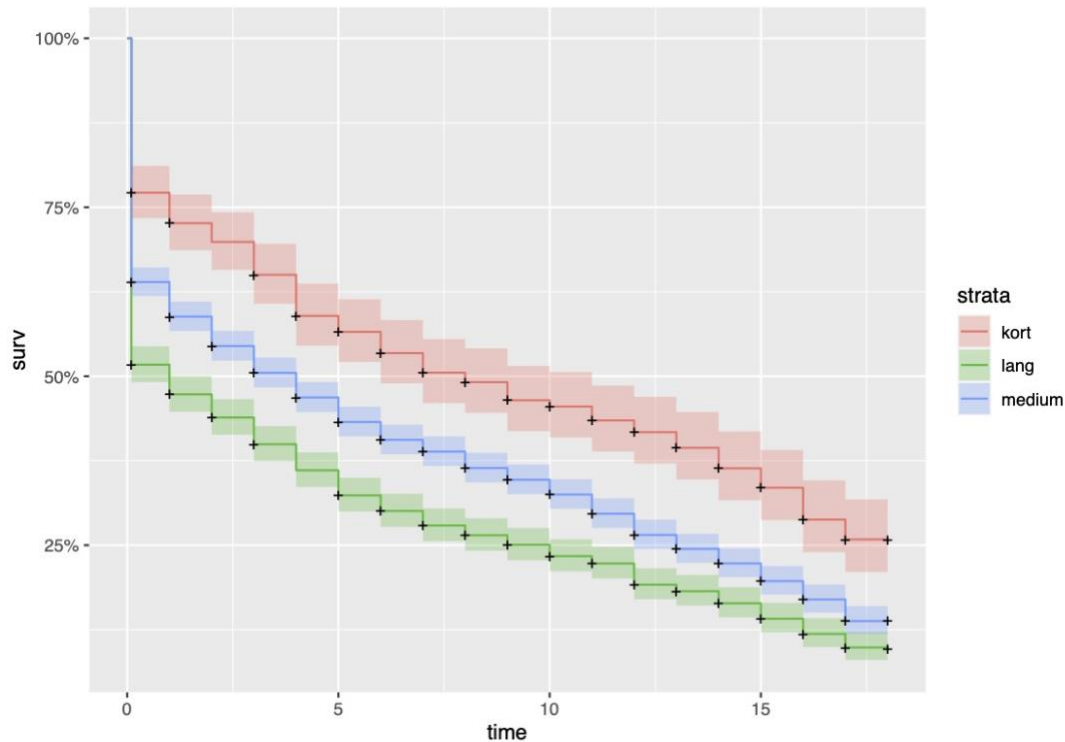


Figuur 13: Survivalcurve met als verklarende variabele "bron".

5.1.1. Fonetische kenmerken

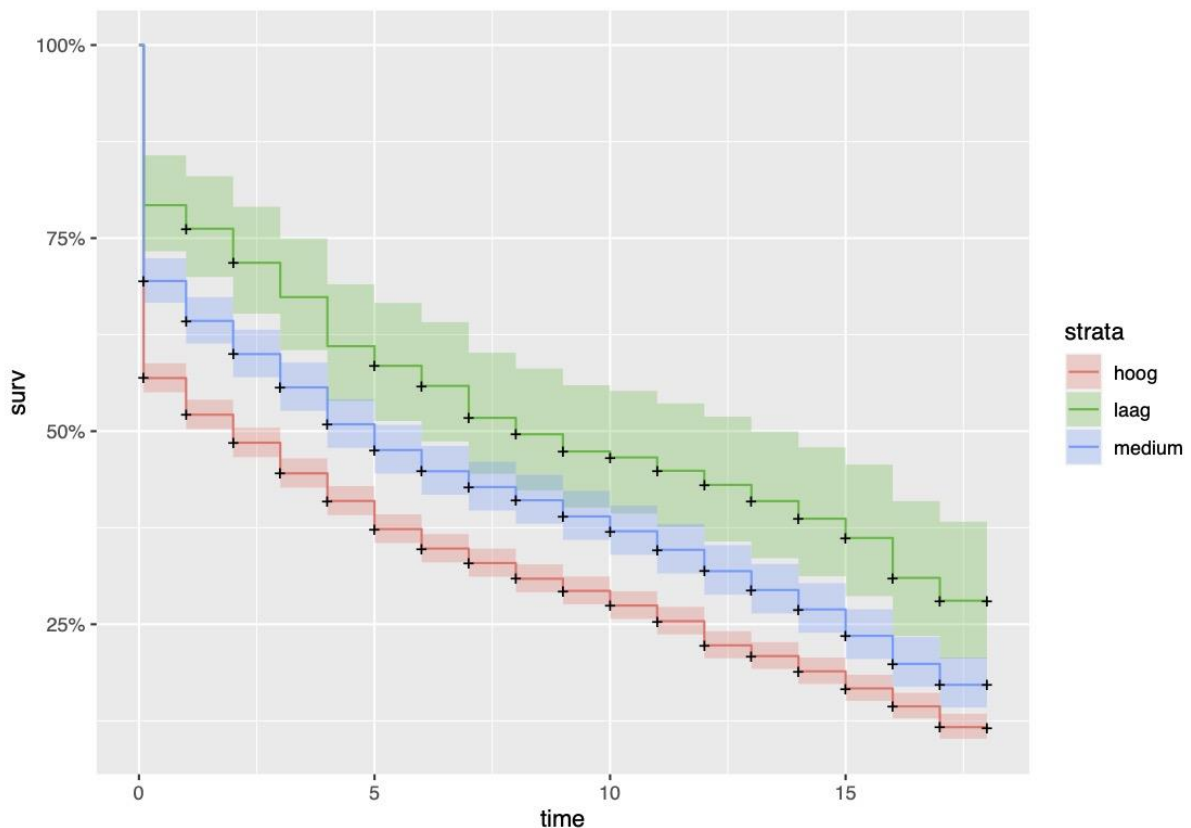
De Kaplan-Meier analyse met als verklarende variabele "lengte" bevestigt de hypothese dat korte woorden een hogere overlevingskans hebben (Van de Velde 2020; Van de Velde & Keersmaekers 2020). Kortere neologismen (lengte=3-8 karakters, N=451) hebben een hogere kans op overleving dan medium (lengte=9-14 karakters, N=1980) of lange (lengte= 15-32 karakters, N=1373) neologismen (Chisq= 101 on 2 degrees of freedom, $p = <0.001$, en ook de pairwise comparisons hebben allemaal $p = <0.001$). Bovendien treedt er dus ook een trend die zich op

macrodiachrone schaal voordoet (Van de Velde 2020) op in microdiachronie. Een verklaring daarvoor is dat kortere woorden makkelijker te onthouden en te leren zijn dan langere.



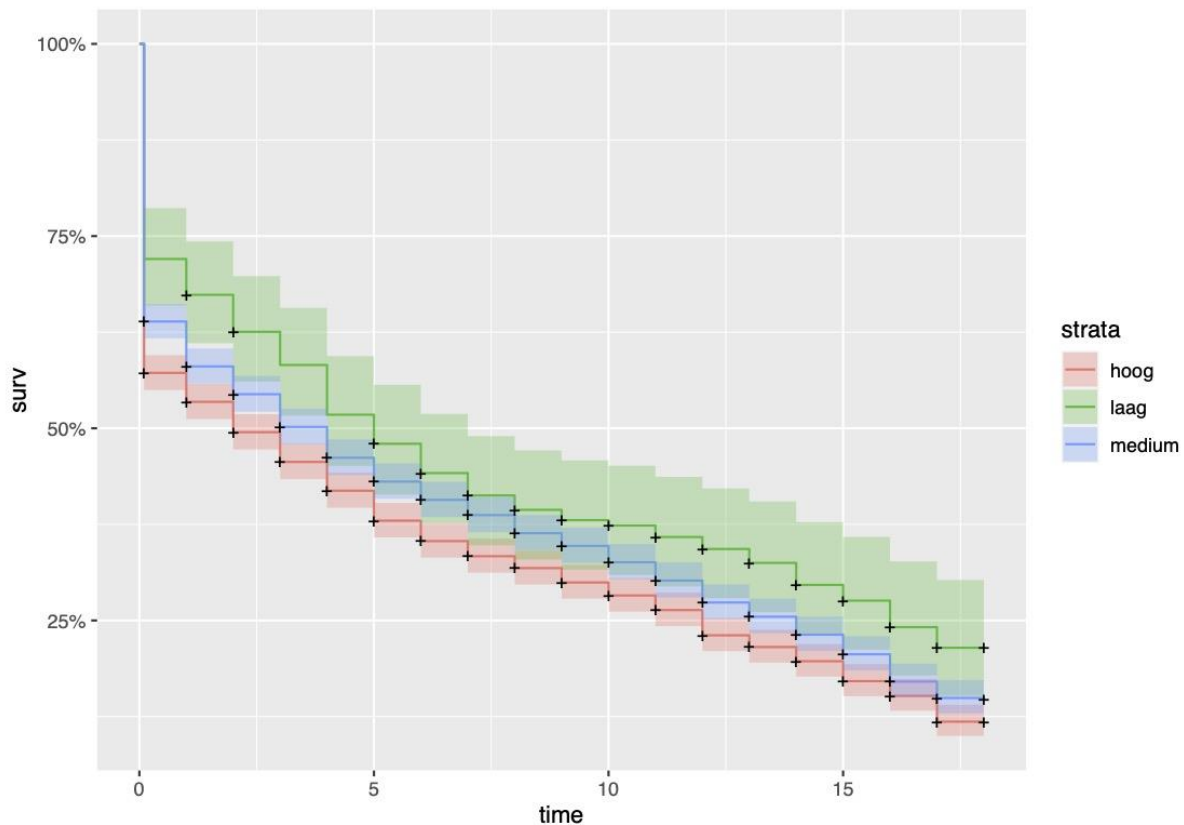
Figuur 14: Survivalcurve met als verklarende variabele "lengte".

Ook de Levenshteinafstand blijkt een bepalende factor in het succes van een neologisme. Neologismen met een lage (afstand=1-2, N=164) Levenshteinafstand tot andere woorden hebben een hogere overlevingskans dan die met een medium (afstand=3-4, N=982) of hoge afstand (afstand=5-26, N=2658). Van de twee concurrerende hypothesen - die van Van de Velde (2020) die stelt dat een lage Levenshteinafstand voordelig is versus die van Metcalf (2002) dat onopvallende woorden een hogere overlevingskans hebben - wint die van Metcalf dus wat betreft deze neologismen. Woorden die sterk lijken op andere woorden en dus onopvallend zijn qua spelling, woordvorming... hebben een hogere kans op overleving. Wel moet worden opgemerkt dat het betrouwbaarheidsinterval van de lage Levenshteinafstand relatief groter is dan die van de hoge en medium Levenshteinafstand. Ze overlappen elkaar echter bijna niet, wat dan toch zorgt voor een redelijk betrouwbaar besluit (Chisq= 61,2 on 2 degrees of freedom, $p = <0.001$, en ook de pairwise comparisons hebben $p = <0.001$, behalve voor laag vs. medium is $p = < 0.01$).



Figuur 15: Survivalcurve met als verklarende variabele "Levenshteinafstand".

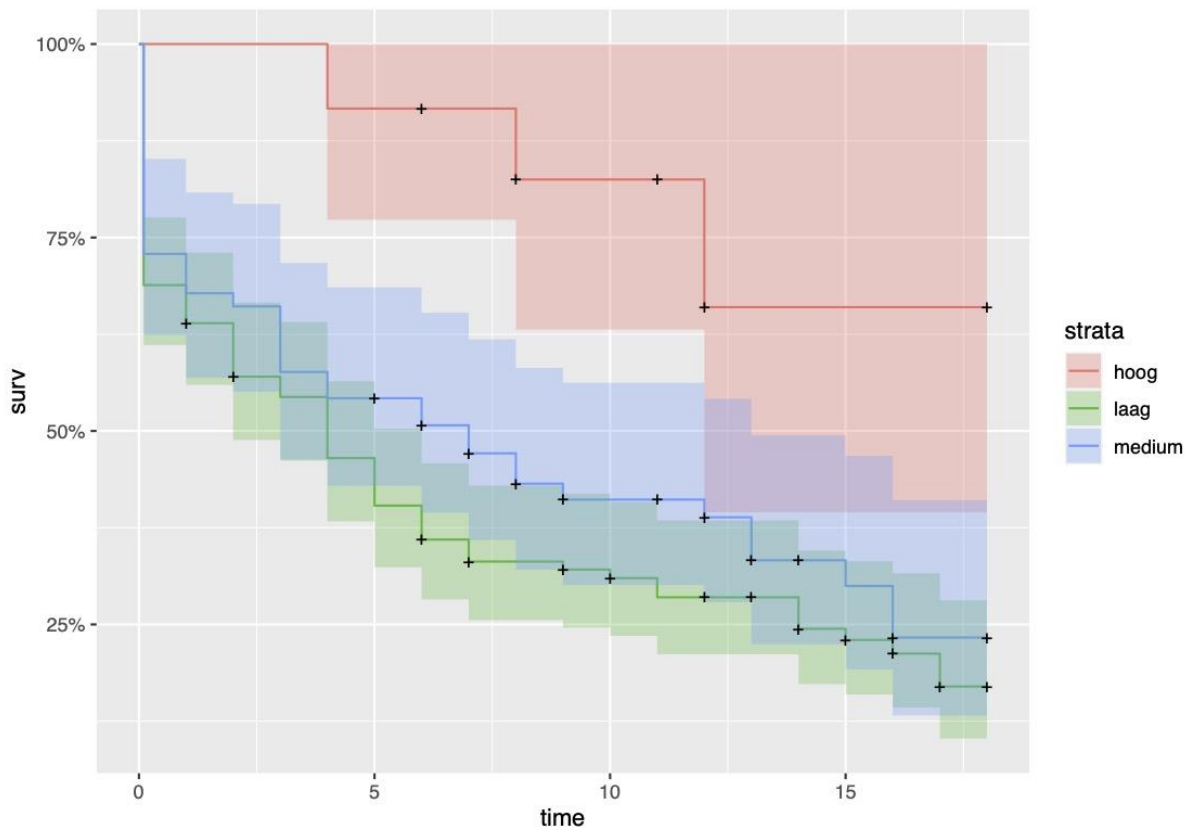
Bij deze grafiek is echter niet de lengte van het neologisme in achtig genomen. Wanneer een lang woord twee ingrepen nodig heeft om een ander woord te worden is dat minder ingrijpend dan wanneer een kort woord twee ingrepen nodig heeft. Daarom werd de gewogen Levenshteinafstand berekend door de Levenshteinafstand te delen door het aantal karakters van het neologisme. Dat resulteerde in 193 neologismen met een lage gewogen Levenshteinafstand (afstand = 0-0,29), 1786 met een medium (afstand = 0,3-0,49) en 1825 met een hoge (afstand = 0,5-1) gewogen Levenshteinafstand. Door de relatieve positie van de curves t.o.v. elkaar te vergelijken, bleek dat neologismen met een lage gewogen afstand een hogere overlevingskans hebben dan neologismen met een medium of hoge gewogen Levenshteinafstand (Chisq = 17,9 on 2 degrees of freedom, $p < 0.001$). Bovendien zijn de pairwise p-waardes significant ($p < 0.05$ voor laag vs. medium en $p < 0.01$ voor hoog vs. medium en hoog vs. laag).



Figuur 16: Survivalcurve met als verklarende variabele "gewogen Levenshteinafstand".

Verdergaand met de neologismen met een lage gewogen Levenshteinafstand (afstand=0-0,29) was er de hypothese dat als een woord erg lijkt op een ander woord dat op zijn beurt heel frequent is, dat de overlevingskans negatief zou kunnen beïnvloeden. Uit de Kaplan-Meier analyse blijkt echter het omgekeerde: neologismen die sterk lijken op hoogfrequente woorden hebben een hogere overlevingskans. Dat gaat in tegen mijn hypothese geïnspireerd door Van de Velde (2020) dat concurreren met een hoogfrequent woord nadelig is omdat mensen de woorden dan met elkaar kunnen verwarren. Uit de curve blijkt echter dat het net een voordeel is om op een hoogfrequent woord te lijken. Dat kan te maken hebben met het feit dat het neologisme dan wellicht voldoet aan de spellingsregels van het Nederlands en dus niet te veel opvalt (Metcalf 2002). Belangrijk is wel dat het aantal data voor deze analyse relatief klein was (N=193), namelijk 12 hoogfrequente (frequentie 5.211-188.542), 59 mediumfrequente (frequentie 600-4.587) en 122 laagfrequente (frequentie 158-591) woorden. Dat zorgt voor grote betrouwbaarheidsintervallen die elkaar overlappen, wat dat besluit minder betrouwbaar maakt. De p-waardes zijn in het algemeen wel

significant (Chisq= 15,1 on 3 degrees of freedom, $p = <0.01$). Ook het verschil tussen hoog en laag is significant: $p = < 0.01$, tussen laag en medium niet en tussen hoog en medium was $p = <0.05$.

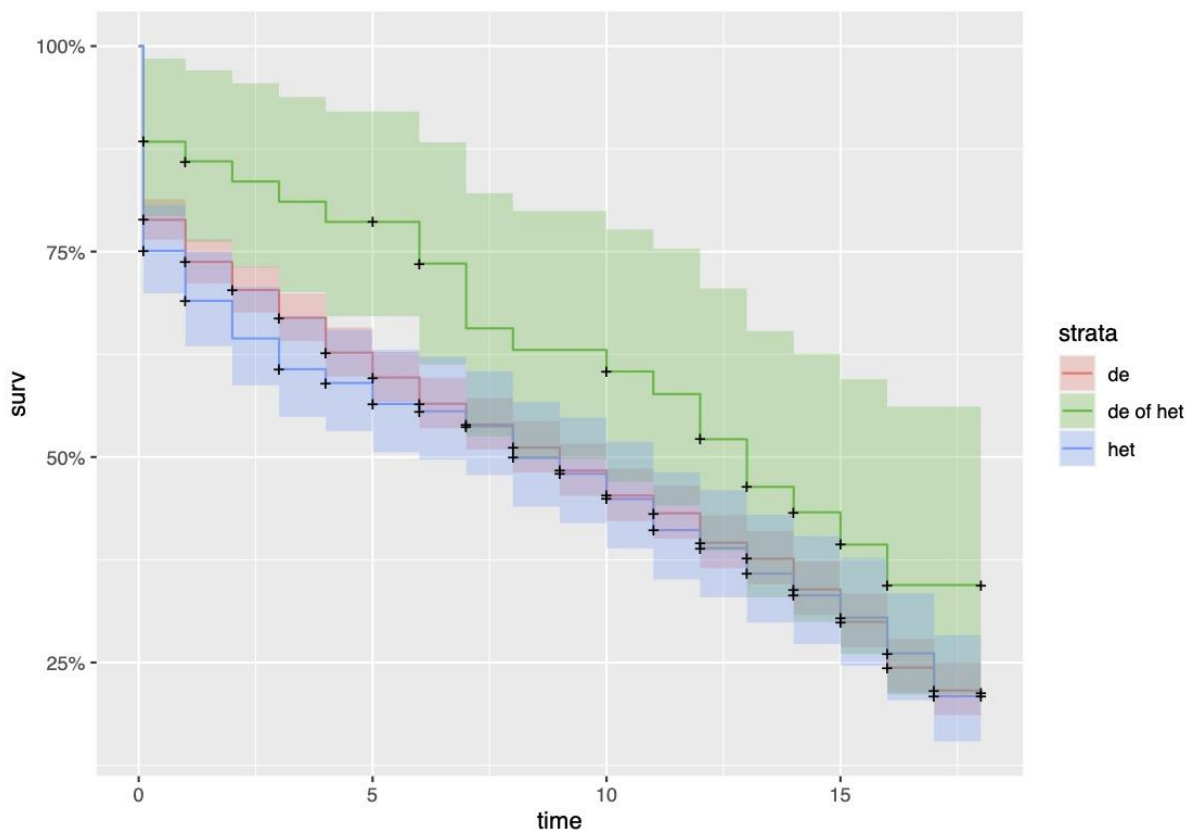


Figuur 17: Survivalcurve met als verklarende variabele "frequentie van het dichtstbijzijnde woord".

5.1.2. Morfologische kenmerken

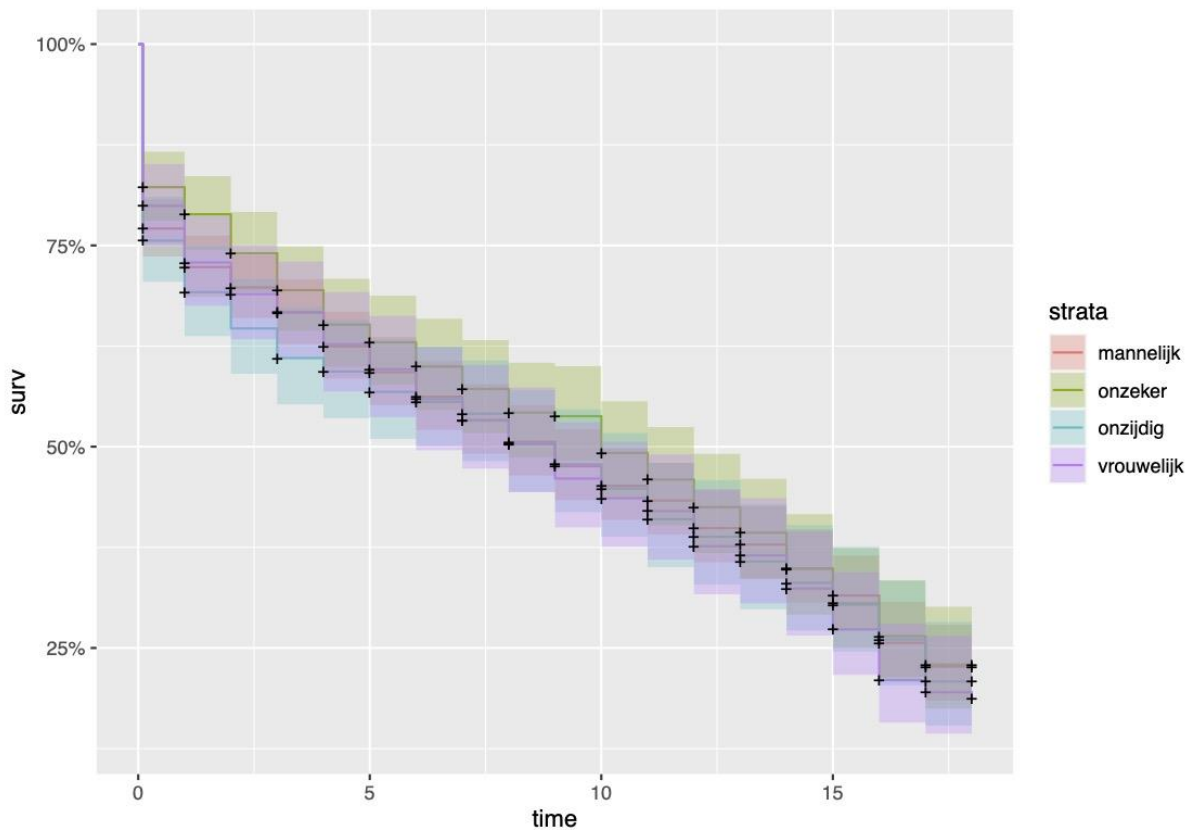
De verklarende variabele "lidwoord" leidt in de Kaplan-Meier analyse niet tot een voordeel of nadeel. Neologismen die als lidwoord "de of het" hebben lijken wel een iets hogere kans op overleving te hebben. Een verklaring daarvoor is dat 23 van de 43 woorden met "de" of "het" als lidwoord een samenstelling met *blog* zijn. *Blog* is een erg productief rechterlid en kan dus zorgen voor een hogere overlevingskans. Woorden met als lidwoord "de" (N=1041) hebben in het begin ook een iets hogere overlevingskans dan woorden met als lidwoord "het" (N=249) (78,9% vs. 75,1%), wat de hypothese dat *het*-woorden een kleinere overlevingskans hebben (Hulk & Cornips 2006; Blom et al. 2008) zou kunnen bevestigen. Uiteindelijk eindigen de *het*-woorden

en *de*-woorden echter op ongeveer dezelfde hoogte. Bovendien zijn de betrouwbaarheidsintervallen en p-waardes redelijk groot en overlappen ze elkaar, wat een besluit bemoeilijkt (Chisq= 348 on 4 degrees of freedom, $p = <0.001$, maar de pairwise comparisons zijn $p = >0.05$ en dus niet significant).



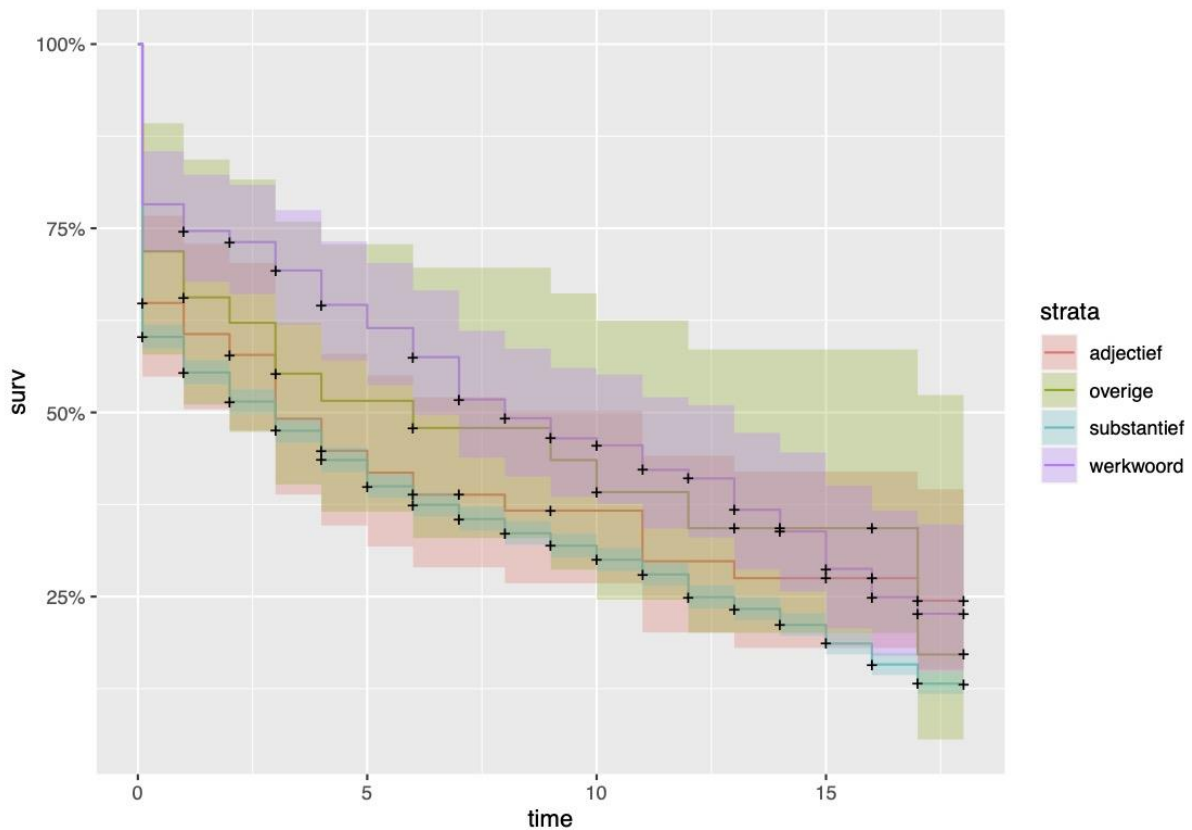
Figuur 18: Survivalcurve met als verklarende variabele "lidwoord".

Ook geslacht blijkt een variabele te zijn die weinig verklaart. De verschillende variabelen - mannelijk (N=537), vrouwelijk (N=244), onzijdig (N=254) en onzeker (N=304) - overlappen en kruisen elkaar en hebben geen significante p-waardes (Chisq= 348 on 5 degrees of freedom, $p = <0.001$, maar $p = >0.05$ voor de pairwise comparisons). Het bevestigt wel al eerder onderzoek (Cornips & De Vogelaer 2009; De Vogelaer 2006) dat het gendersysteem in het Nederlands stilaan irrelevant wordt en zou kunnen verdwijnen.



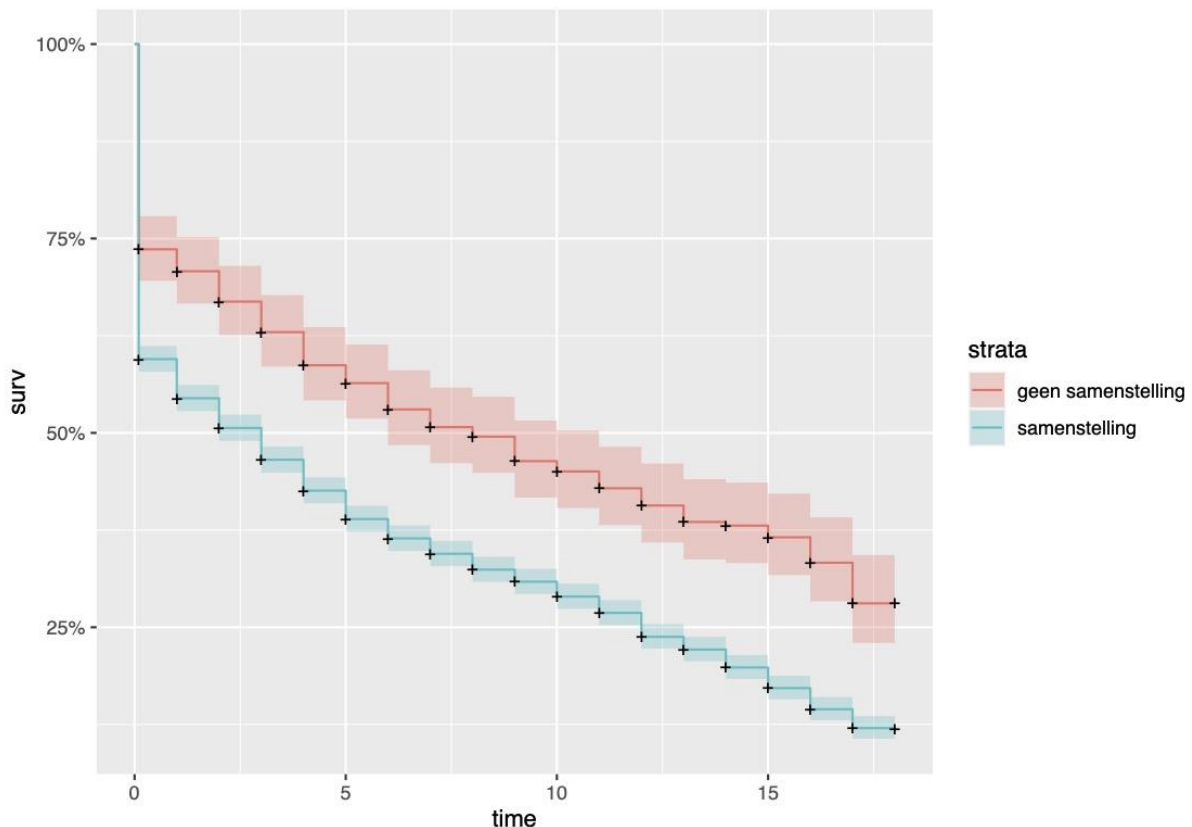
Figuur 19: Survivalcurve met als verklarende variabele "geslacht".

De Kaplan-Meier curve met als verklarende variabele "woordsoort" bevestigt enigszins de hypothese die voortkomt uit het onderzoek van Van de Velde (2020) en Van de Velde & Keersmaekers (2020) dat adjectiva minder succesvol zijn dan nomina en verba, waarbij verba ook nog succesvoller zijn dan nomina. De verba (N=138) hebben een hogere overlevingskans dan de nomina (N=3560) en de adjectiva (N=74) in de dataset, weliswaar met overlappende betrouwbaarheidsintervallen ($\text{Chisq} = 19,9$ on 3 degrees of freedom, $p = <0.001$). Zeker in het begin zien we dat van de werkwoorden 78,3% overleeft, tegenover 60,3% van de nomina en 64,9% van de adjectiva. Het significantste verschil zien we tussen substantieven en werkwoorden ($p = <0.001$). Ook hier komt dus een fenomeen dat op macrodiachronie voorkomt ook in microdiachronie naar boven, namelijk de hogere overleving voor werkwoorden. De overige woorden (N=32) vormen een te kleine groep om een betrouwbare uitspraak over te doen.



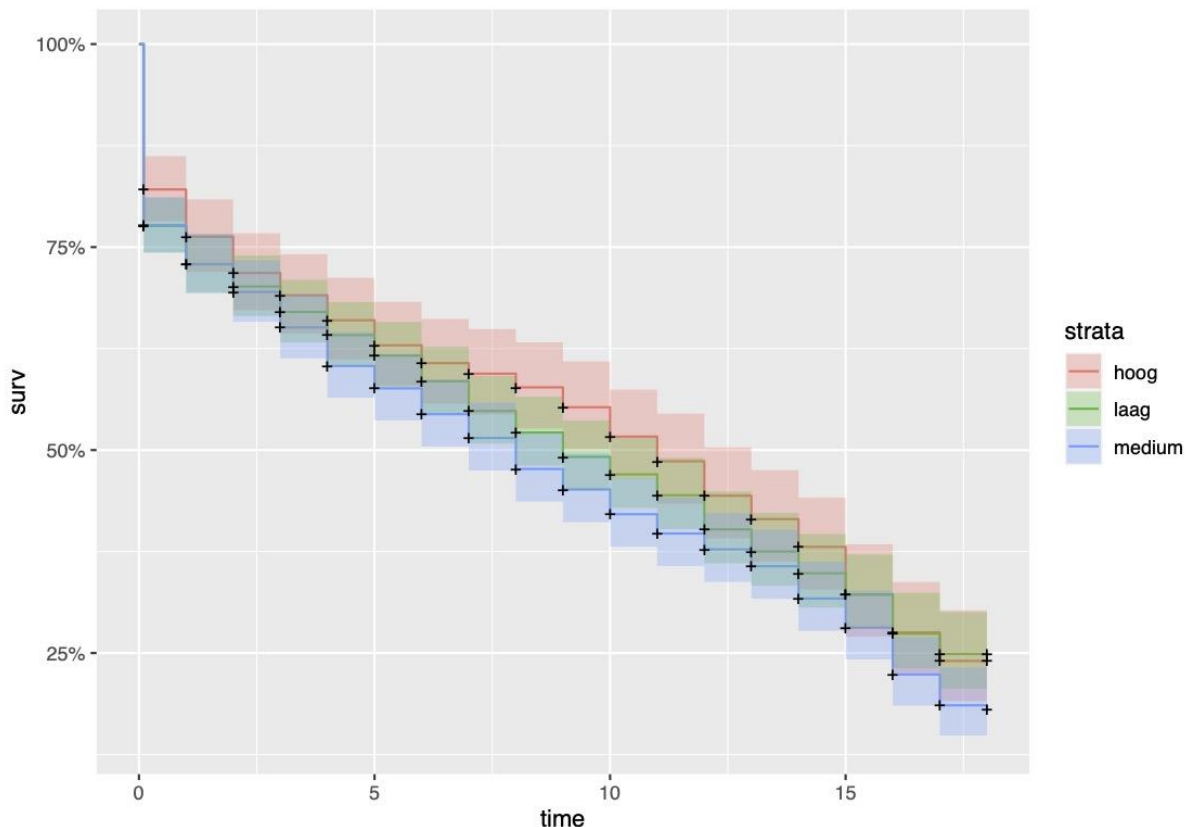
Figuur 20: Survivalcurve met als verklarende variabele "woordsoort".

Wanneer we de woordvorming van de neologismen in twee splitsen: "samenstellingen" (N=3372) en "niet-samenstellingen" (N=432) is het opmerkelijk dat de Kaplan-Meier analyse een hogere overlevingskans geeft voor "niet-samenstellingen" (Chisq= 58,7 on 1 degrees of freedom, $p < 0.001$). Dat gaat in tegen de hypothese dat samenstellingen succesvoller zouden zijn omdat ze meestal bestaan uit herkenbare delen (Waszink 2016). Ongeveer 168 niet-samenstellingen zijn leenwoorden, 109 inheems en de rest is onzeker. De curve lijkt dan ook erg op die met als variabele "leenwoord". Een mogelijke verklaring voor het feit dat niet-samenstellingen langer leven is dat ze over het algemeen korter zijn dan samenstellingen (gemiddeld 9,6 vs. 13,8 karakters), en uit een eerdere curve bleek al dat kortere woorden een langer leven beschoren zijn.



Figuur 21: Survivalcurve met als verklarende variabele "samenstelling".

Verdergaand met die 3372 samenstellingen was de hypothese geïnspireerd op Metcalfs (2002) criterium "generation of forms and meanings" en Renoufs levensfase "productiviteit" (2013) dat als de samenstelling bestaat uit delen die op zich erg productief zijn en vaak voorkomen in andere samenstellingen, dat de overlevingskans omhoog zou kunnen trekken omdat de herkenbaarheid van het woord dan ook hoger is. De hoogfrequente samenstellende delen (frequentie 3014-234.886, N=352) hebben in het begin een betere overlevingskans in vergelijking met de mediumfrequente (frequentie 752-2997, N=574) en laagfrequente (frequentie 0-736, N=585) samenstellende delen (82,1% vs. 77,7% en 77,6%). Op het einde hebben de laagfrequente echter een hogere overlevingskans, met weliswaar een miniem verschil, overlappende betrouwbaarheidsintervallen en kruisende curves van hoog en laag, waardoor de curves eigenlijk niet goed interpreteerbaar zijn (Chisq= 349 on 3 degrees of freedom, $p = <0.001$, maar de pairwise comparison was enkel significant voor hoog vs. medium met $p = <0.05$).

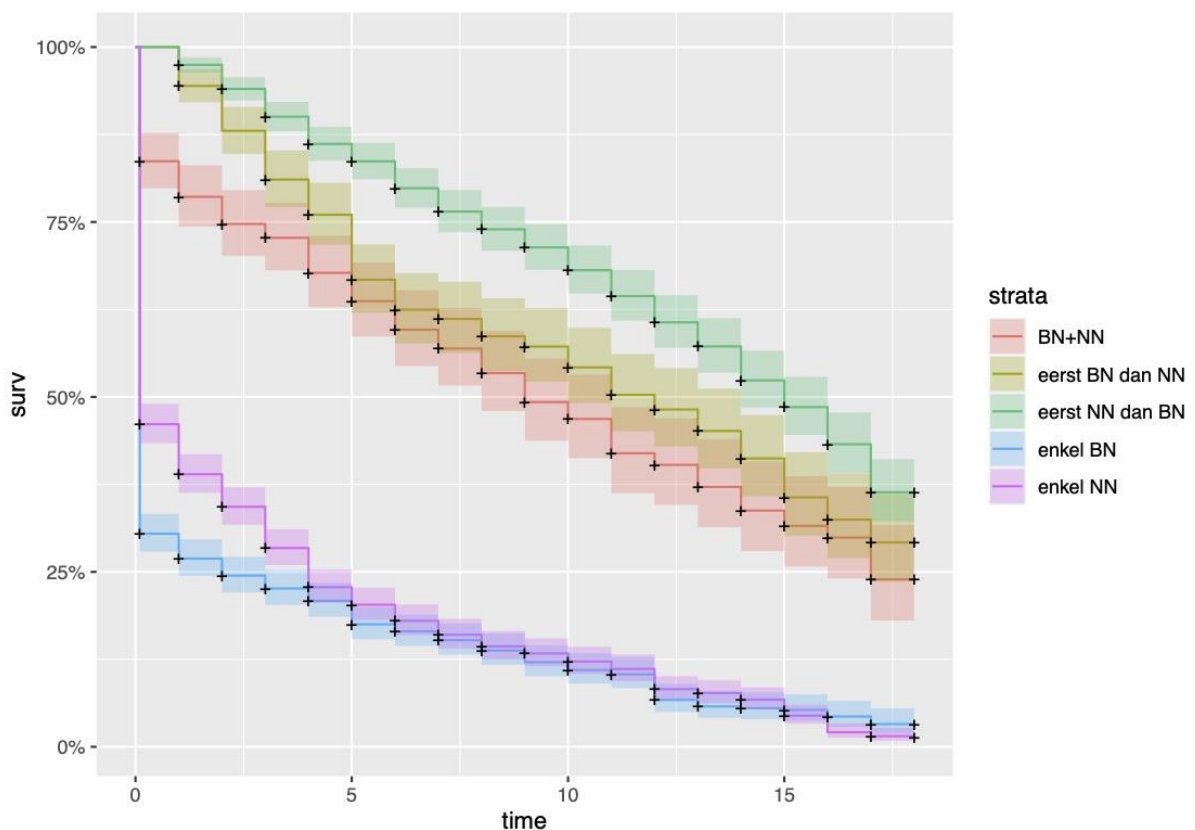


Figuur 22: Survivalcurve met als verklarende variabele "productiviteit van de samenstellende delen".

5.1.3. Sociolinguïstische en cognitief-linguïstische kenmerken

De diffusie of de verspreiding van de neologismen tussen de twee variëteiten van het Nederlands zorgt voor een opvallend verschil in de Kaplan-Meier curve. Enerzijds zijn er de neologismen die in twee variëteiten voorkomen: hetzij tegelijk (BN+NN, N=337), of eerst BN dan NN (N=360), of eerst NN dan BN (N=786). Anderzijds zijn er de neologismen die in slechts één variëteit voorkomen: enkel BN (N=1113) of enkel NN (N=1208). Uit die tweedeling blijkt dat de neologismen die in twee variëteiten voorkomen een hogere overlevingskans hebben dan die in één variëteit opduiken (Chisq= 1462 on 4 degrees of freedom, $p < 0.001$), wat de pairwise p-waardes ook bevestigen ($p < 0.001$, behalve voor eerst BN dan NN vs. BN+NN en voor enkel NN vs. enkel BN is de p-waarde < 0.05). Zeker in het begin is er een groot verschil: 97,5% van eerst NN dan BN overleeft zijn eerste jaar, tegenover 30,4% van enkel BN. Belangrijk daarbij te vermelden is wel dat de curves van enkel BN en enkel NN elkaar kruisen en daarna praktisch samenvallen. Taalcontact, of in dit geval variëteitcontact,

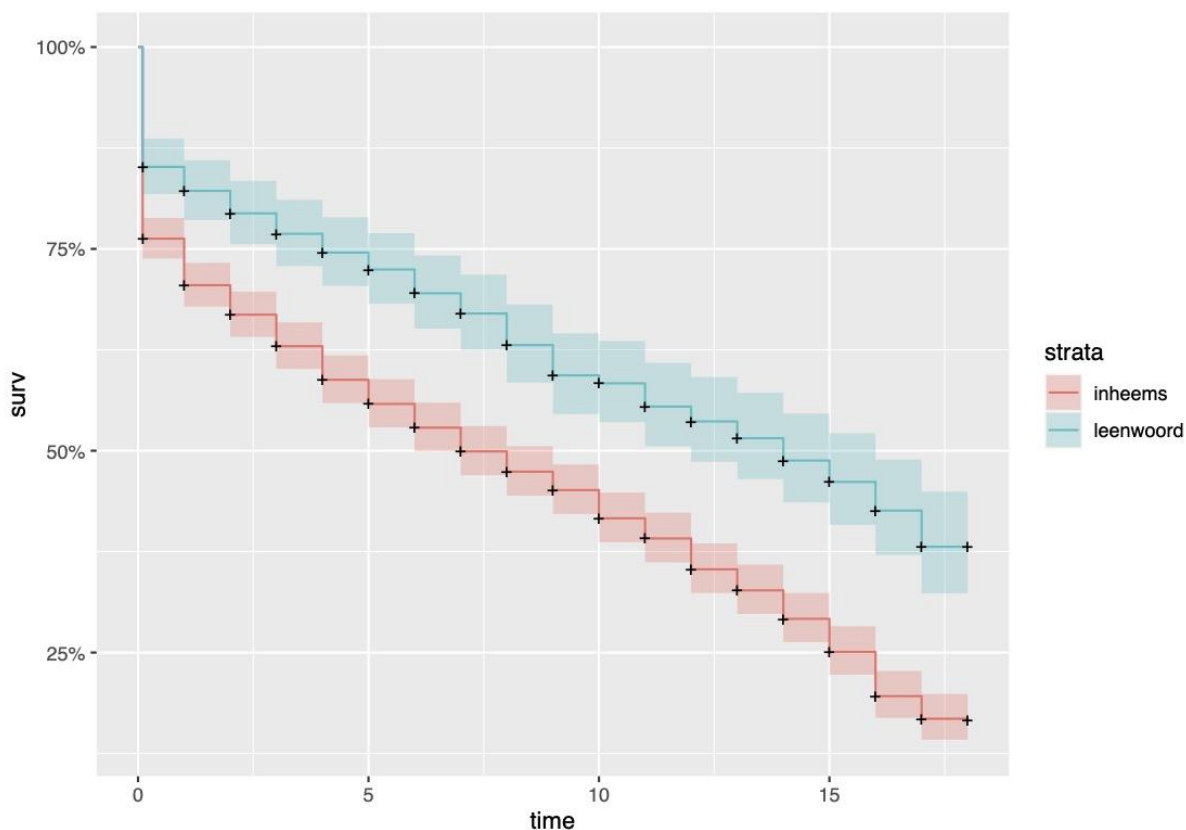
blijkt dus een belangrijk gegeven in de overlevingskans van neologismen. De hoogste overlevingskans heb je als neologisme als je eerst in het NN voorkomt en dan in het BN, waaruit de grotere invloed van het NN op de Nederlandse woordenschat blijkt. Deze curve bevestigt dus de hypothese dat diffusie zorgt voor een betere overlevingskans, wat voortkomt uit het onderzoek van Würschinger (2021) en Zenner et al. (2012).



Figuur 23: Survivalcurve met als verklarende variabele "diffusie".

Zoals al eerder vermeld lijkt de curve van samenstelling versus geen samenstelling erg op de curve van inheems (N=1100) versus leenwoord (N=411). Leenwoorden, vaak blends uit het Engels en geen samenstellingen, hebben een hogere overlevingskans (Chisq= 381 on 2 degrees of freedom, $p < 0.001$) dan inheemse woorden. Dat bevestigt de hypothese dat als men het de moeite vindt om een woord uit een andere taal te lenen, het woord dan ook wel zal overleven en succesvol zal zijn. De drempel om een woord uit een andere taal te lenen is in het begin hoger dan om zelf een woord te vormen in de eigen taal. Wanneer die drempel dan toch

overschreden is, is het wel de moeite waard het woord te blijven gebruiken (Zenner et al. 2012).

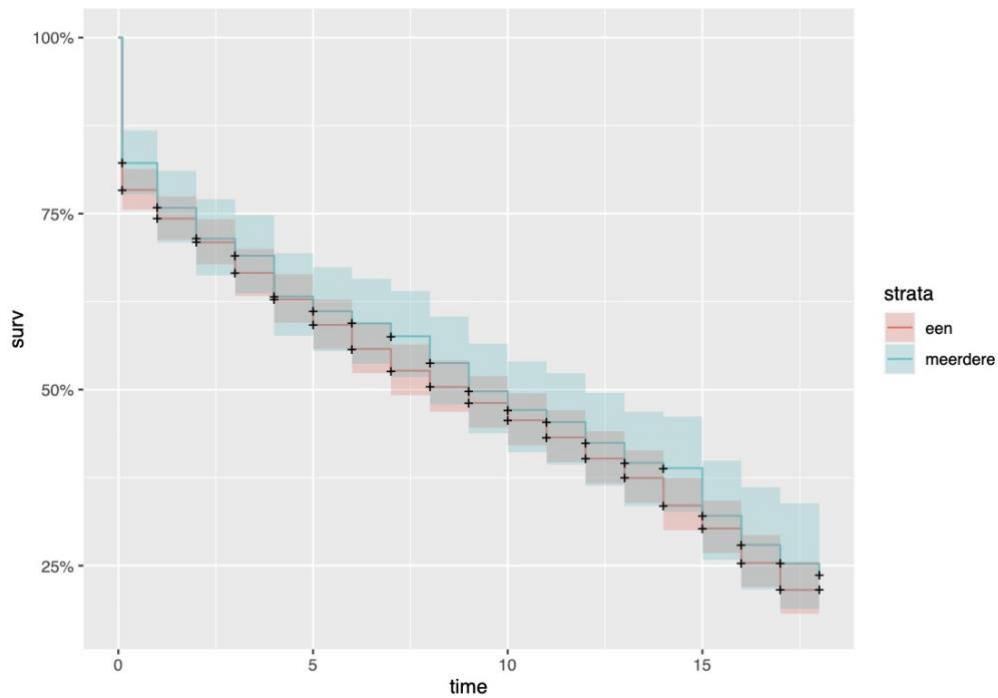


Figuur 24: Survivalcurve met als verklarende variabele "herkomst".

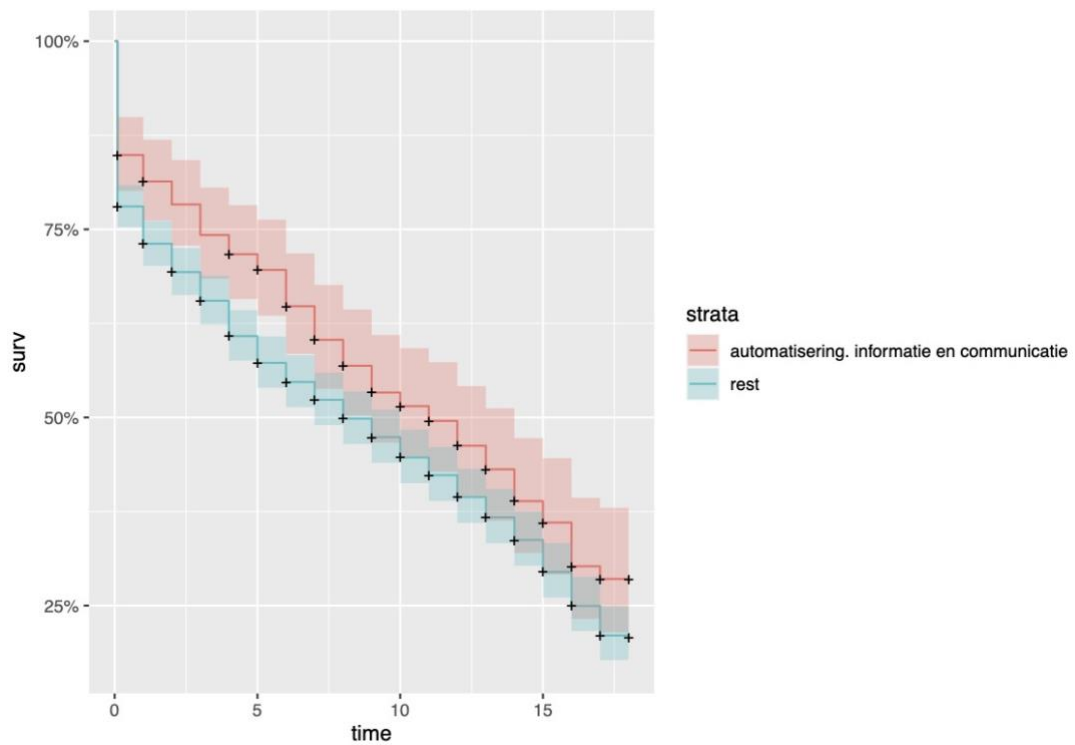
5.1.4. Semantische kenmerken

Wanneer men kijkt naar neologismen die door het WNW in een domein worden ingedeeld (N=777), versus de neologismen die in meerdere domeinen worden gecategoriseerd (N=275) is er amper verschil qua overlevingskans (Chisq= 183 on 2 degrees of freedom, $p = <0.001$, maar $p = > 0.05$ bij de pairwise comparison). De hypothese van Metcalf (2002) die zegt dat "diversity of users and situations" zorgt voor een goede overlevingskans kan met deze dataset dus niet bevestigd worden. Wanneer men echter de neologismen opdeelt in het vaakst voorkomende domein, "automatisering, informatie en communicatie" (N=205) en de rest (N=847), blijkt dat dat vaakst voorkomende domein wel zorgt voor succesvollere en langer overlevende neologismen (Chisq= 185 on 2 degrees of freedom, $p = <0.001$, en $p = <0.05$ bij de pairwise comparison), wat de hypothese was (Waszink 2016). Woorden uit het domein "automatisering, informatie en communicatie" zijn ook vaak leenwoorden uit het

Engels, waarvan hierboven (Figuur 24) al bewezen is dat ze een hogere overlevingskans hebben.

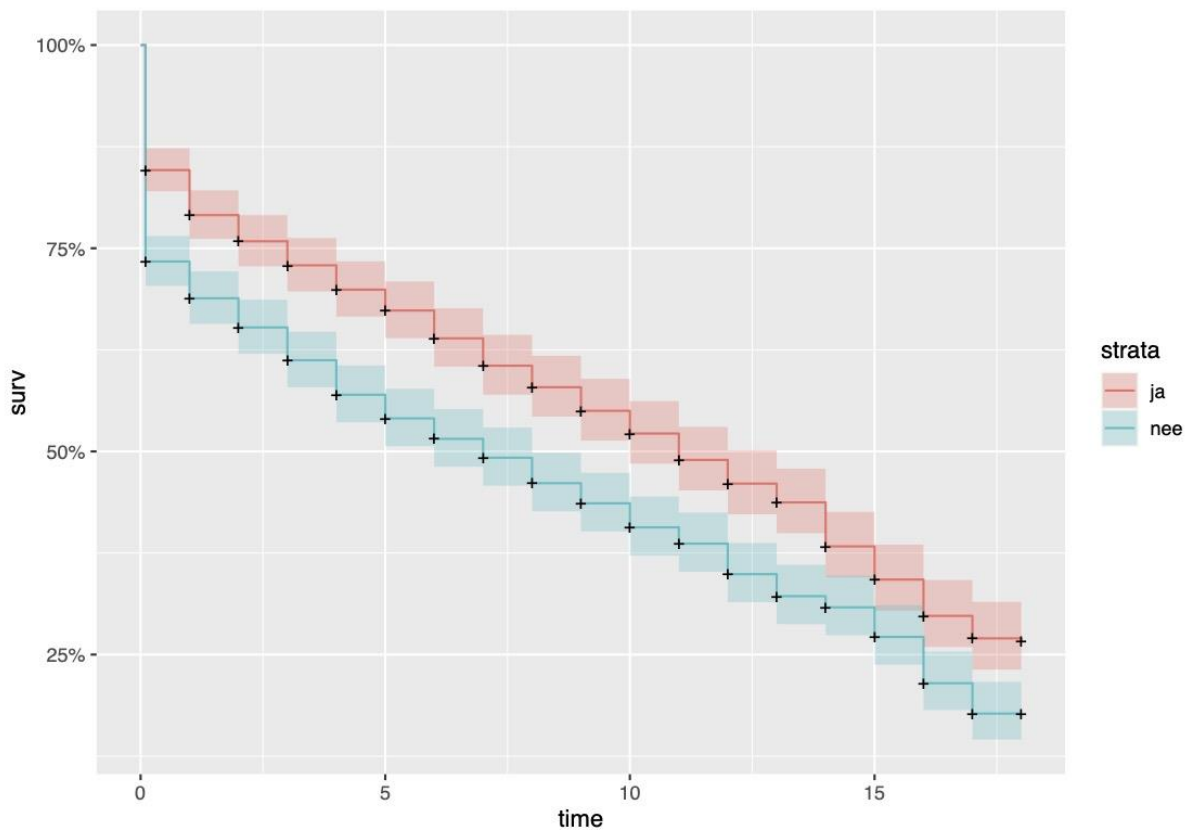


Figuur 25: Survivalcurve met als verklarende variabele "een of meer domeinen".



Figuur 26: Survivalcurve met als verklarende variabele "automatisering, informatie en communicatie".

Wat betreft synonymie was de hypothese dat wanneer een neologisme moet concurreren met een bestaand woord, het een lagere overlevingskans heeft (Zenner et al. 2012). Uit de Kaplan-Meier analyse blijkt echter het tegenovergestelde. Woorden die één of meer synoniemen hebben (ja, N=715) hebben een hogere overlevingskans (Chisq= 359 on 2 degrees of freedom, $p = <0.001$ en ook bij de pairwise comparison is $p = <0.001$) dan neologismen die er geen hebben (nee, N=796). Een verklaring daarvoor is dat als een woord een synoniem heeft, het concept dat dat woord aanduidt al goed ingeburgerd is en dus ook al een tijd overleefd heeft. Daarvoor zou in vervolgonderzoek nog onderzocht kunnen worden of de leeftijd van het concept een invloed heeft (Zenner et al. 2012).



Figuur 27: Survivalcurve met als verklarende variabele "synonymie".

5.2. Cox Proportional Hazards model

Een logische volgende stap in dit onderzoek is om over te stappen van een bivariaat model (de Kaplan-Meier) naar een multivariaat model: het Cox Proportional Hazards model. Terwijl een Kaplan-Meier slechts één verklarende variabele onderzoekt, kan het Cox Proportional Hazards model de invloed van meerdere variabelen in kaart brengen. Dat is vooral relevant voor variabelen die in de Kaplan-Meier curves hetzelfde patroon vertonen. Met een Cox Proportional Hazards model kunnen die twee variabelen dan vergeleken worden en uitsluitsel geven of de variabelen hetzelfde effect of een apart effect teweegbrengen. Bovendien geeft het model ook inzicht in hoe groot het effect van de variabelen op de overleving is (Van Geloven 2018). Dit deel staat echter onder voorbehoud; een Cox Proportional Hazards model heeft voor alle variabelen data nodig, wat niet het geval is in deze dataset. Er kon dus enkel een Cox Proportional Hazards model opgesteld worden voor de variabelen die zowel in het WNW als in *Neoloog* aanwezig waren. Dat zijn de variabelen lengte, woordsoort, samenstelling, diffusie, bron en gewogen Levenshteinafstand. Het Cox Proportional Hazards model laat ook toe voor een aantal variabelen van een ordinaal meetniveau (de indeling kort, medium, lang of hoog, medium, laag voor lengte en Levenshteinafstand) overstappen naar een ratio meetniveau. Voor de analyses werd gebruik gemaakt van het programma *R* (R Core Team 2020).

Voor het opstellen van het Cox Proportional Hazards model heb ik aan manuele *backward model selection* gedaan. In een eerste testmodel werden daarbij alle variabelen in het model gestoken om te kijken welke variabelen de grootste invloed hadden en welke de kleinste (zie Tabel 1). Op basis van dat testmodel werden de variabelen "gewogen Levenshteinafstand" en "woordsoort" als eerste verwijderd. Uit het model blijkt namelijk dat hun p-waardes niet erg significant zijn.

Tabel 1: Cox Proportional Hazards model met alle variabelen.

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ gewogen +
      Samenstelling + Woordsoort + aantal.karakters + bron + Diffusie,
      data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)
gewogen	-0.544435	0.580170	0.215410	-2.527	0.011490 *
Samenstellingsamenstelling	0.099783	1.104931	0.072646	1.374	0.169579
Woordsoortoverige	0.254741	1.290128	0.264255	0.964	0.335047
Woordsoortsubstantief	0.070404	1.072941	0.144453	0.487	0.625988
Woordsoortwerkwoord	0.081431	1.084839	0.179687	0.453	0.650417
aantal.karakters	0.013741	1.013836	0.005057	2.717	0.006579 **
bronwnw	-0.613843	0.541267	0.045104	-13.609	< 2e-16 ***
Diffusieeerst BN dan NN	-0.372611	0.688933	0.101420	-3.674	0.000239 ***
Diffusieeerst NN dan BN	-0.591397	0.553554	0.089547	-6.604	3.99e-11 ***
Diffusieenkel BN	1.064892	2.900526	0.082234	12.950	< 2e-16 ***
Diffusieenkel NN	1.095134	2.989583	0.079388	13.795	< 2e-16 ***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
gewogen	0.5802	1.7236	0.3804	0.8849
Samenstellingsamenstelling	1.1049	0.9050	0.9583	1.2740
Woordsoortoverige	1.2901	0.7751	0.7686	2.1655
Woordsoortsubstantief	1.0729	0.9320	0.8084	1.4241
Woordsoortwerkwoord	1.0848	0.9218	0.7628	1.5428
aantal.karakters	1.0138	0.9864	1.0038	1.0239
bronwnw	0.5413	1.8475	0.4955	0.5913
Diffusieeerst BN dan NN	0.6889	1.4515	0.5647	0.8404
Diffusieeerst NN dan BN	0.5536	1.8065	0.4644	0.6598
Diffusieenkel BN	2.9005	0.3448	2.4688	3.4078
Diffusieenkel NN	2.9896	0.3345	2.5588	3.4929

Concordance= 0.786 (se = 0.005)

Likelihood ratio test= 1748 on 11 df, p=<2e-16

Wald test = 1485 on 11 df, p=<2e-16

Score (logrank) test = 1724 on 11 df, p=<2e-16

Een tussenstap was nakijken of de variabelen "samenstelling" en "aantal karakters" hetzelfde verklaren. Samenstellingen zijn zoals al eerder vermeld gemiddeld langer dan niet-samenstellingen, dus rijst er de vraag of aantal karakters en samenstelling niet hetzelfde effect hebben, wat blijkt uit de Kaplan-Meier curves. Wanneer die twee variabelen in een apart Cox model worden gestoken, blijkt dat er

toch nog een apart effect is. Dit verschil had echter geen hoge c-waarde ($c=0.584$, $se=0.007$).⁷ Om die reden wordt er maar een van de twee variabelen behouden in het model, namelijk aantal karakters omdat samenstelling in het testmodel niet significant was.

De variabelen met de grootste significantie zijn "diffusie" en "bron". Uit de Kaplan-Meier van "diffusie" blijkt dat de waarde "initieel NN" een hogere overlevingskans geeft. Het WNW heeft ook meer initieel NN-neologismen dan BN. De vraag is dan of het WNW ook succesvoller is dan *Neoloog* door die bias naar initieel NN-neologismen. Wanneer we deze twee variabelen in een Cox model steken blijkt dat er een effect van "bron" op "diffusie" komt ($c=0.78$, $se=0.005$). Naast het feit dat eerst in het NN voorkomen en dan in het BN erg voordelig is voor neologismen, is dan ook nog eens opgenomen zijn in het WNW een recept voor succes. De lexicografen van het WNW voegen met hun expertise dus nog iets extra's toe dat niet in het model zit.

Wanneer we nu die twee variabelen combineren in een model (zie Tabel 2) met "aantal karakters" krijgen we een significant model ($p= <0.001$) met een vrij goede verklaringsgraad ($c=0.787$, $se=0.005$), die het hoogste is in vergelijking met een ander model (samenstelling + diffusie + bron had een c-waarde van 0.781, $se=0.005$, er is dus slechts een klein verschil). Hieruit blijkt ook dat de *binning* van de variabele "lengte" in "kort", "medium" en "lang" niet optimaal was, aangezien de waardes stijgen wanneer er overgestapt wordt van een ordinaal naar een ratio meetniveau ($c=0.782$ vs $c=0.787$).

Samengevat laat dit model zien dat korte neologismen, die een diffusie ondergaan - en dan vooral van NN naar BN - en waarover de lexicografen van het WNW hebben beslist ze op te nemen in het WNW een hogere overlevingskans hebben. Er moest echter geroeid worden met de riemen die we hadden: niet alle variabelen zijn meegenomen in de analyse wegens het ontbreken ervan in *Neoloog*. Bovendien is er ook niet nagekeken of er interacties tussen variabelen waren. Zo is het mogelijk dat het effect van lengte niet hetzelfde is in *Neoloog* versus het WNW. Bovendien blijkt uit het model dat de lexicografen van het WNW nog een extra variabele hanteren die de overlevingskans van neologismen positief beïnvloedt.

⁷ Zie Appendix V voor de Coxtabellen.

Tabel 2: Cox Proportional Hazards model met variabelen "bron", "diffusie" en "aantal karakters".

```

Call:
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Diffusie +
      bron + aantal.karakters, data = wnw)

n= 3804, number of events= 2877

              coef exp(coef) se(coef)      z Pr(>|z|)
Diffusieeerst BN dan NN -0.371311  0.689829  0.101156 -3.671 0.000242 ***
Diffusieeerst NN dan BN -0.586835  0.556084  0.089334 -6.569 5.06e-11 ***
Diffusieeenkel BN        1.071393  2.919445  0.081765 13.103 < 2e-16 ***
Diffusieeenkel NN        1.108482  3.029756  0.078901 14.049 < 2e-16 ***
bronwnw                   -0.591402  0.553551  0.043893 -13.474 < 2e-16 ***
aantal.karakters          0.009945  1.009995  0.004403  2.258 0.023917 *
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

              exp(coef) exp(-coef) lower .95 upper .95
Diffusieeerst BN dan NN  0.6898      1.4496    0.5658    0.8411
Diffusieeerst NN dan BN  0.5561      1.7983    0.4668    0.6625
Diffusieeenkel BN        2.9194      0.3425    2.4872    3.4269
Diffusieeenkel NN        3.0298      0.3301    2.5957    3.5365
bronwnw                   0.5536      1.8065    0.5079    0.6033
aantal.karakters          1.0100      0.9901    1.0013    1.0187

Concordance= 0.787 (se = 0.005 )
Likelihood ratio test= 1739 on 6 df,  p=<2e-16
Wald test               = 1481 on 6 df,  p=<2e-16
Score (logrank) test = 1718 on 6 df,  p=<2e-16

```

De variabele "bron" is dan ook nog eens subjectief, terwijl de andere variabelen objectief zijn. De subjectiviteit van de variabele "bron" kon dus niet helemaal weggemodelleerd worden. Wat dat extra iets is dat de lexicografen toevoegen is dus zeker stof voor vervolgonderzoek. Een hypothese is alvast dat de variabele "herkomst" (leenwoord vs. inheems) er voor iets tussen kan zitten. Misschien heeft het WNW meer leenwoorden opgenomen dan *Neoloog* en zorgt dat voor een betere overleving. Als alle verklarende variabelen in het model zaten, zou de variabele "bron" dus geen effect meer hebben. Ook niet alle modeldiagnostieken zijn uitgevoerd. Zo is er geen expliciete test van heterogeniteit afgenomen die rekening houdt met effecten die niet constant zijn doorheen de tijd, zoals diffusie. Bij sommige neologismen kan die namelijk sneller of trager gebeuren.

6. Conclusies

Het doel van dit onderzoek was bepalen welke variabelen het succes en de levensduur van neologismen beïnvloeden. Daarbij kwam ook de vraag of survivalanalyses beter uitsluitsel kunnen geven dan frequentieverlopen. Het merendeel van de neologismen sterft al in zijn eerste levensjaar, waardoor de frequentieverlopen van deze dataset niet erg relevant zijn. Wanneer we echter overstappen op een minder traditionele neologismenanalyse - Kaplan-Meier en Cox Proportional Hazards modellen - krijgen we een beter zicht op welke variabelen bepalend zijn, aangezien survivalanalyses rekening houden met de neologismen die onderweg "overlijden". Belangrijk is dat de dataset heel anders is dan vorig evolutionair-taalkundig onderzoek. Hier gaat het om woorden die net geboren zijn in het Nederlands, terwijl die uit Van de Velde & Keersmaekers (2020) en Van de Velde (2020) al langer in het Griekse of Nederlandse lexicon zitten. Dat verklaart het grote verschil in het verloop van de Kaplan-Meier curve. Geëtableerde woorden ondergaan een langzame afsterving, tegenover de massale initiële sterfte bij neologismen. Ondanks dat grote verschil is het wel opmerkelijk dat er enkele verklarende variabelen zorgen voor een gelijkaardig effect bij beide datasets.

Zo konden we de hypothesen over de meeste verklarende variabelen bevestigen. In het algemeen kan men zeggen dat een groot aantal neologismen slechts een kort leven beschoren is. Een eerste factor die echter zorgt voor een hogere overlevingskans heeft te maken met woordenboekopname. De neologismen die in het WNW staan zijn succesvoller dan die uit *Neoloog*, wat betekent dat de lexicografen van het WNW goed voorspellende criteria hanteren. Deze studie kan echter helpen om een nog betere inschatting te maken van welke neologismen nu "blijvertjes" zijn.

Een volgende verklarende variabele waarvan de hypothese bevestigd is, is "lengte". Kortere neologismen hebben een hogere overlevingskans dan lange, een patroon dat ook op macrodiachronie voorkomt. Hetzelfde kan gezegd worden over "woordsoort", waarbij zowel op micro- als macrodiachronie verba een hogere overlevingskans hebben dan nomina en adjectiva. Neologismen die een lage Levenshtein afstand hebben en dus onopvallend zijn, hebben ook een hogere overlevingskans. Dat is dan weer het omgekeerde van wat op macrodiachrone schaal gebeurt. Een andere variabele waarbij de hypothese net omgekeerd bleek, is

woordvorming. Neologismen die geen samenstellingen zijn - en dus vaak leenwoorden zijn - zijn een langer leven beschoren dan samenstellingen. Dat wordt ook bevestigd door de survivalanalyse van de variabelen "leenwoord vs. inheems". Samenstellingen met erg productieve samenstellende delen hebben dan weer wel een hoge overlevingskans, weliswaar met een miniem verschil.

Uit de analyse van de variabele "diffusie" blijkt de relatief grotere invloed van het Nederlands-Nederlands op de rest van het taalgebied. Neologismen die een vorm van verspreiding ondergaan zijn succesvoller, wat de hypothese bevestigt. Of een neologisme in een of meer domeinen gebruikt wordt, blijkt geen doorslaggevende variabele. Wanneer men echter naar het meest voorkomende domein, "automatisering, informatie en communicatie" kijkt, zorgt dat wel voor een hogere overlevingskans. Dat hangt ook weer samen met het feit dat zulke woorden vaak leenwoorden zijn, die sowieso een hogere overlevingskans hebben.

Neologismen met meerdere synoniemen hebben een hogere kans op overleving in vergelijking met neologismen zonder synoniemen. Dat spreekt de hypothese tegen dat de neologismen dan in concurrentie staan met woorden die mogelijk al langer in het lexicon zitten. Het is echter wel logisch dat als een woord een synoniem heeft, het concept dat het woord aanduidt goed ingeburgerd is en dus langer zal kunnen overleven. Wat de variabelen "lidwoord" en "geslacht" betreft kunnen er geen al te betrouwbare uitspraken gedaan worden, daarvoor zou meer data onderzocht moeten worden.

Bij deze conclusies kunnen echter enkele vragen gesteld worden, aangezien er slechts één variabele tegelijkertijd wordt onderzocht. Samenstellingen, bijvoorbeeld, zijn uit zichzelf redelijk lang, dus is het plausibel dat de variabele "hoog aantal karakters" eigenlijk samenvalt met de variabele "samenstelling". Een andere vraag is of het WNW succesvoller is dan *Neoloog* door die bias naar initieel NN-neologismen die blijkt uit de diffusie-analyse. Daarom werd een Cox Proportional Hazards model opgesteld dat met meerdere variabelen tegelijk rekening kan houden. Daaruit bleek dat de variabelen "samenstelling" en "aantal karakters" toch een apart effect hebben op neologismen. Ook bij "bron" en "diffusie" is er een apart effect van bron bovenop diffusie. Het Cox Proportional Hazards model maakt duidelijk dat de variabelen "samenstelling", "woordsoort", en "gewogen Levenshteinafstand" geen significant verschil maken wanneer ze samen met de andere variabelen vergeleken

worden. Het zijn dus vooral de variabelen "bron", "diffusie" en "aantal karakters" die de grootste bepalende factoren zijn van het succes en de levensduur van neologismen.

Om deze resultaten betrouwbaarder en beter interpreteerbaar te maken, zou er in vervolgonderzoek met een grotere dataset gewerkt kunnen worden. Door het gebrek aan tijd konden niet alle variabelen manueel toegevoegd worden aan de neologismen uit *Neoloog*, met name "herkomst" lijkt een belangrijke ontbrekende variabele. Als dat in de toekomst verwezenlijkt zou kunnen worden, zou er een beter Cox Proportional Hazards model opgesteld kunnen worden. Ook zouden er nog andere variabelen bekeken kunnen worden, enkele suggesties zijn: de productiviteit van het neologisme zelf, dus hoe vaak het neologisme zelf gebruikt wordt om een andere samenstelling/afleiding... te vormen. Er zou ook rekening gehouden kunnen worden met derivationele morfologie bij de Levenshtein afstand, aangezien niet alle woorden die op elkaar lijken zorgen voor verwarring. Enkel processen waarbij begripsverwarring of ongewenste synonymie wordt veroorzaakt zouden dus onderscheiden moeten worden van woorden die geen herkeningsprobleem hebben. De opmerking bij de resultaten voor de variabele "synonymie" was dat de inburgering van het concept misschien invloed heeft. Om dat zeker te weten zou er gekeken kunnen worden naar de leeftijd van het concept (zie Zenner et al. 2012). Ook andere soorten van diffusie zijn interessant om te onderzoeken, bijvoorbeeld de verspreiding van neologismen over katernen in kranten of tussen gebruikers op Twitter. Verder zou ook de mate waarin een nieuw woord leeft, of de frequentieverlopen, geïntegreerd kunnen worden in de survivalanalyses als tijdsvariërende covariant. Zo zou onderzocht kunnen worden of de frequentie van neologismen een invloed heeft op de overlevingskans. Die variabele is bewust niet opgenomen in dit onderzoek omdat die frequenties veranderlijk zijn en dus de analyse zou compliceren. Een extra analysemethode die nog toegepast zou kunnen worden op deze data is een Random Forest. Op die manier zou er een beslissingsboom opgesteld kunnen worden die de lexicografen van neologismenwoordenboeken kunnen hanteren als inclusiemethode.

7. Bibliografie

- Blom, W.B.T., Polisenska, D. & Weerman, F. (2008). Articles, adjectives and age of onset: the acquisition of Dutch grammatical gender. *Second Language Research*, 24(3), 297–331. <https://doi.org/10.1177/0267658308090183>.
- Cartier, E. (2017). Neoveille, a Web Platform for Neologisme Tracking. Proceedings of the EACL 2017 Software Demonstrations, 95-98. <https://doi.org/10.18653/v1/E17-3024>.
- Cornips, L. & De Vogelaer, G. (2009). Variatie en verandering in het Nederlandse genus: een multidisciplinair perspectief. *Taal En Tongval*, 61(1), 1–12. <https://doi.org/10.5117/TET2009.1.CORN>.
- Corpus Hedendaags Nederlands - CHN (Versie 3.0) (Oktober 2021) [Online Service]. Beschikbaar bij het Instituut voor de Nederlandse Taal: <http://hdl.handle.net/10032/tm-a2-s8>. (Geraadpleegd op 30/12/2021).
- De Vogelaer, G. (2006). Pronominaal genus bij “Zuid-Nederlandse” taalverwervers: van grammaticaal naar semantisch systeem. In *Nederlands tussen Duits en Engels* (pp. 89–102). Stichting Neerlandistiek Leiden.
- Falk, I. Bernhard, D. & Gérard, C. (2018). *The Logoscope: a Semi-Automatic Tool for Detecting and Documenting French New Words*.
- Freixa, J. & Torner, S. (2020). Beyond Frequency: On the Dictionarization of New Words in Spanish. *Dictionaries: Journal of the Dictionary Society of North America* 41(1), 131-153. [doi:10.1353/dic.2020.0008](https://doi.org/10.1353/dic.2020.0008).
- Grieve, J., Nini, A. & Guo, D. (2017). Analyzing lexical emergence in Modern American English online. *English Language and Linguistics*, 21(1), 99–127. <https://doi.org/10.1017/S1360674316000113>.
- Horikoshi, M. & Tang, Y. (2018). *ggfortify: Data Visualization Tools for Statistical Analysis Results*. <https://CRAN.R-project.org/package=ggfortify>.
- Hulk, A. & Cornips, L. (2006). Neuter Gender and Interface Vulnerability in Child L2/2L1 Dutch. In *PATHS OF DEVELOPMENT IN L1 AND L2 ACQUISITION: IN HONOR OF BONNIE D. SCHWARTZ*, Unsworth, Sharon, Parodi, Teresa, Sorace, Antonella, & Young-Scholten, Martha [Eds], Amsterdam, NE: John Benjamins, 2006, pp 107-134.

- Humboldt, W. Von. (2011). *Über die Verschiedenheit des menschlichen Sprachbaues: und ihren Einfluß auf die geistige Entwicklung des Menschengeschlechts*. Cambridge University Press.
- INT ANW = Instituut voor de Nederlandse Taal (2021). *Over het ANW*. <https://anw.ivdnt.org/about> (Geraadpleegd op 30/12/2021).
- INT CHN = Instituut voor de Nederlandse Taal (2021). *About the Corpus Hedendaags Nederlands (CHN)*. <https://portal.clarin.inl.nl/corpus-frontend-chn/chnextern/about> (Geraadpleegd op 30/12/2021).
- INT WNW = Instituut voor de Nederlandse Taal (2021). *Over het Woordenboek van Nieuwe Woorden*. <https://neologismen.ivdnt.org/about> (Geraadpleegd op 30/12/2021).
- Janssen, M. (2008). *NeoTrack: Un analyseur de néologismes en ligne*. In M.T. Cabré, O. Domènech, R. Estopà & J. Freixa (eds.) *Proceedings of CINEO 2008*, pp 1175-1188.
- Jiang, M., Shen, X. Y., Ahrens, K. & Huang, C.-R. (2021). Neologisms are epidemic: Modeling the life cycle of neologisms in China 2008-2016. *PloS One*, 16(2), e0245984–e0245984. <https://doi.org/10.1371/journal.pone.0245984>.
- Kassambara, A., Kosinski, M. & Biecek, P. (2019). *survminer: drawing survival curves using 'ggplot2'*. R package version 0.4.6. <https://CRAN.R-project.org/package=survminer>.
- Kehoe, A. & Gee, M. (2009). Weaving web data into a diachronic corpus patchwork. In *Corpus Linguistics* (Vol. 69, pp. 255–279). https://doi.org/10.1163/9789042025981_015.
- Kerremans, D., Stegmayr, S. & Schmid, H.-J. (2011). The NeoCrawler: identifying and retrieving neologisms from the internet and monitoring ongoing change. In *Current Methods in Historical Semantics* (Vol. 73, pp. 59–96). DE GRUYTER. <https://doi.org/10.1515/9783110252903.59>.
- Kerremans, D. (2015). A Web of New Words. A Corpus-Based Study of the Conventionalization Process of English Neologisms. *English Corpus Linguistics* 15.
- Kerremans, D. & Prokić, J. (2018). Mining the Web for New Words: Semi-Automatic Neologism Identification with the NeoCrawler. *Anglia (Tübingen)*, 136(2), 239–268. <https://doi.org/10.1515/ang-2018-0032>.

- Keuleers, E., Brysbaert, M. & New, B. (2010). SUBTLEX-NL: A new measure for Dutch word frequency based on film subtitles. *Behavior Research Methods*, 42(3), 643–650. <https://doi.org/10.3758/BRM.42.3.643>.
- Klosa-Kückelhaus, A. & Wolfer, S. (2020). Considerations on the Acceptance of German neologisms from the 1990s. *International Journal of Lexicography*, 33(2), 150–167. <https://doi.org/10.1093/ijl/ecz033>.
- Kosem, I., Krek, S., Gantar, P., Holdt, S.A. & Čibej, J. (2021). Language Monitor: Tracking the Use of Words in Contemporary Slovene. *Electronic Lexicography in the 21st Century (ELex 2021) Post-Editing Lexicography*, 514-528.
- Metcalf, A. (2002). *Predicting New Words. The Secrets of their Success*. Houghton Mifflin.
- Nimb, S., Sørensen, N. H. & Lorentzen H. (2020). Updating the dictionary: Semantic change identification based on change in bigrams over time. *Slovenščina 2.0*, 8(2), 112-138. <https://doi.org/10.4312/slo2.0.2020.2.112-138>.
- OLD20 = Keuleers, E. (2010). Fast computation of average Levenshtein distances. <http://crr.ugent.be/programs-data/fast-computation-of-average-levenshtein-distances-in-python-including-old20>. (Geraadpleegd op 31/05/2022).
- Pollak, S., Gantar, P. & Arhar Holdt, Š. (2019). What's New on the Internet? Extraction and Lexical Categorisation of Collocations in Computer-Mediated Slovene. *International Journal of Lexicography*, 32(2), 184–206. <https://doi.org/10.1093/ijl/ecy026>.
- R Core Team (2020). R: A language and environment for statistical computing. *R Foundation for Statistical Computing*, Vienna, Austria. <https://www.R-project.org/>.
- Renouf, A. (1993). A Word in Time: first findings from dynamic corpus investigation. In J. Aarts, P. de Haan, & N. Oostdijk (eds.) *English Language Corpora: Design, Analysis and Exploitation*, 279-288.
- Renouf, A. (2009). Corpus Linguistics beyond Google: the WebCorp Linguist's Search Engine. *Digital Studies*, 1(1). <https://doi.org/10.16995/dscn.138>
- Renouf, A. (2013). A finer definition of neology in English: The life-cycle of a word. *Studies in Corpus Linguistics*, 57, 177–207. <https://doi.org/10.1075/scl.57.14ren>.

- Schmid, H. (2008). New Words in the Mind: Concept-formation and Entrenchment of Neologisms. *Anglia (Tübingen)*, 126(1), 1–36.
<https://doi.org/10.1515/angl.2008.002>.
- Therneau, T. (2022). A Package for Survival Analysis in R. R package version 3.3-1,
<https://CRAN.R-project.org/package=survival>.
- Trap-Jensen, L. (2020). Language-Internal Neologisms and Anglicisms: Dealing with New Words and Expressions in *The Danish Dictionary*. *Dictionaries: Journal of the Dictionary Society of North America* 41(1), 11-25.
[doi:10.1353/dic.2020.0002](https://doi.org/10.1353/dic.2020.0002).
- Van der Sijs, N. (2002). *Chronologisch woordenboek: De ouderdom en herkomst van onze woorden en betekenissen* (2de dr. ed.).
- Van de Velde, F. & Keersmaekers, A. (2020). What are the determinants of survival curves of words? An evolutionary linguistics approach. *Evolutionary Linguistic Theory*, 2(2), 127–137. <https://doi.org/10.1075/elt.00019.vel>.
- Van de Velde, F. (2020). Zwindende Woorden. *Op je Woorden Letten - 200 Jaar Matthias de Vries*. Instituut voor de Nederlandse taal.
<https://www.facebook.com/ivdnt/videos/3274010056059652> (Geraadpleegd op 30/12/2021).
- Van Geloven, N. & Geskus, R.B. (2018). Survival analyse. *Wikistatistiek*. Amsterdam UMC. https://wikistatistiek.amc.nl/index.php/Survival_analyse#Kaplan_Meier_analyse (Geraadpleegd op 08/02/2021).
- Veale, T. (2006). Tracking the Lexical Zeitgeist with Wikipedia and WordNet. *In Proceedings of ECAI'2006, the 17th European Conference on Artificial Intelligence*, 56-60.
- Waszink, V. (2016). Kraamkost, tabletbochels en lelijke groente: hoe zien nieuwe woorden eruit en wie gebruikt ze? *Ons erfdeel*, 59(3), 26-32.
- Waszink, V. (2020). Neologisms in an Online Portal: The Dutch *Neologismenwoordenboek* (NW). *Dictionaries: Journal of the Dictionary Society of North America* 41(1), 27-44. [doi:10.1353/dic.2020.0003](https://doi.org/10.1353/dic.2020.0003).
- Wickham, H. (2016). *ggplot2: Elegant Graphics for Data Analysis*. Springer-Verlag, New York.

- WNW = Instituut voor de Nederlandse Taal (2022). Woordenboek van Nieuwe Woorden. <https://ivdnt.org/woordenboeken/nieuwe-woorden/>. (Geraadpleegd op 19/05/2022).
- Würschinger, Q. (2021). Social Networks of Lexical Innovation. Investigating the Social Dynamics of Diffusion of Neologisms on Twitter. *Frontiers in Artificial Intelligence*, 4, 648583–648583. <https://doi.org/10.3389/frai.2021.648583>.
- Zenner, E., Speelman, D. & Geeraerts, D. (2012). Cognitive Sociolinguistics meets loanword research: Measuring variation in the success of anglicisms in Dutch. *Cognitive Linguistics*, 23(4), 749–792. <https://doi.org/10.1515/cog-2012-0023>.

8. Appendix

Appendix I: Neologismen uit het WNW

Behouden neologismen uit het WNW (N=1511)

06-dealer, 1,5 metereconomie, 1,5 metersamenleving, 1 euromaaltijd, 10-10-10, 20 wekenecho, 24 uursluier, 2.42-omroep, 3D-echo, 3D-echografie, 3D-monitor, 3D-printer, 3D-schaats, 419-fraude, 45 kilometerwagen, 45 kilometerwagentje, 45km-wagentje, aanblijfpremie, aanbodgestuurd, aandachtsjunk, aandelenleaseproduct, aankoopfraude, aanlandplicht, aanloopleren, aanmodderfakker, aanrechtsubsidie, aardappelgeweer, aardgasrotonde, aardgastaxi, aarsgewei, aasgierfonds, accordeonfile, accudag, achtbaancursus, actieradiusangst, adoniscomplex, adoptieagent, adoptieboom, adoptiekerstboom, adoptiekie, adware, aed-apparaat, aed-toestel, afluistercamera, afschermplicht, afspiegelingscoalitie, afspiegelingsregering, afstandsangst, afstandschaamte, afstandshamen, afvinkcultuur, agecoin, agendahedonist, agendaseks, agflatie, agressiecoach, agropark, agroproductiepark, aidocratie, airsoftwapen, alcoholband, alcoholbus, alcoholpoli, alcoholpolikliniek, alcoholpop, alcoholvrijmibo, alcolock, alerteringssysteem, alleenschaamte, all-inhotel, alt-A-hypotheek, altcoin, alt-light, alt-lite, alt-rechts, alt-right, aluhoedje, ambulancedrone, ambulanceklever, amfibiebus, amoledschem, anderhalvemetereconomie, anderhalvemeteren, anderhalvemetermaatschappij, anderhalvemeterregel, anderhalvemetersamenleving, anticonceptiepleister, anticorruptiewaakhond, antispamfilter, antivaxer, appbericht, appen, appgroep, applauskind, appstore, appwinkel, aquariuskind, artnapping, asbaktegel, asielcriminaliteit, asieltsunami, aso-azc, asobak, asocontainer, asodorp, asomobilist, asotaks, asowoning, aspergerobot, atheïstenkerk, atoomtoerisme, attentioexia, audioblog, audiodescriptie, autipas, autismedcoach, autismeklas, autistenklas, autoklonen, autokloon, autotsaar, baarmoederstaking, baarschaamte, babyluik, babyschuif, bachelor-masterstructuur, badhairday, bajesplan, bakfietsdelen, bakfietsgezin, bakfietskind, bakfietsmoeder, bakfietsouder, bakfietsstad, bakfietsvader, balansbord, balansdag, balanseconomie, baliehuwelijk, balkenendenorm, balkonsolidariteit, bambiseks, banenbonus, banenjager, bankbelasting, bankbus, bankenbelasting, bankendomino, bankenheffing, bankentaks, bankhangen, bankheffing, barebacken, barebacker, barplassen, barplasser, basejumper, basketbaldiplomatie, batterijstress, beachbabe, beachsoccer, beautycoach, bedrijfscoach, begrotingsravijn, begrotingtsaar, beleeftuin, beleidsdiarree, beleidsintimiteit, beleveniscultuur, belevenismarketing, beleveniswinkel, belevingseconomie, belevingsmarketing, belevingsrestaurant, belevingstuin, bel-me-nietregister, belparkeren, benadelingswaan, berm bom, bermbomaanslag, bermmonument, bestuursobesitas, beswaffelen, betaalmuur,

bevallingscoach, beweegtuin, beweeg-vmbo, bewegingscoach, bewijsteam, beyoncéfeminisme, beyoncévlieg, bezorgschaamte, bibdate, bibdaten, biechtsite, bierkeet, bierschip, big data, big tech, bijenmakelaar, bijensnelweg, bijzondere bromfiets, bikinibabe, bingedrinken, binge-eten, bingewatchen, bingewatcher, bingewatching, biofortificatie, biogasbus, biologisch paspoort, bioterrorist, biovergister, biovergisting, birdnesting, bitchfight, bitcoin, bitcoinmiljonair, bladkorf, blaftaks, blaricumeffect, blauwe energie, blindencommentator, blindengeleidepaard, blindengeleidepony, blindentribune, blockchain, bloeddiamant, bloemenpiet, bloggen, blogger, blogosfeer, blokhypotheek, blokjesverjaardag, blokstrooien, blootscanner, blowverbodsbord, bobbycam, bodyshaming, boeddhaknuffel, boekenasiel, boemerangkind, boerkaban, boerkaverbod, boerkini, bokitoproof, Bollywoodfilm, bonusbelasting, bonusplafond, booslim, boosterlening, bore-out, borstvoedingsbank, borstvoedingsvriendelijk, bosbrossen, bosbrosser, boshuwelijk, boterhammentaks, botnet, boycotgroente, brambulance, breezer, breezerseks, breindoping, breinpil, Brexit, Brexodus, broeikastoerisme, broekzakbellen, broekzakbeller, bromance, brooddoosregeling, bubbelbadbioscoop, bucketlist, budgetcoach, budgetkliniek, budgetluchtvaartmaatschappij, budgetspreekuur, buigtelefoon, buikmoeder, buikstoot, buitenaula, buitenrijpas, bumpersurfen, bumpersurfer, burgerhulpverlener, burn-outpatiënt, bushcraft, bushcrafter, buttkicker, buurtbibliotheek, buurtvader, bvz, camerajournalist, cameraval, camperaar, campingzender, cancelcultuur, cannabispil, cannachopper, cardiobingo, cardiostick, carrotmob, casinopensioen, catchphrase, caviapolitie, caxirola, censuurteam, CEO-fraude, cheerleadereffect, chemobom, cheugy, chillaxen, chloorkip, christenhipster, chronischevermoeidheidssyndroom, circulaire carrière, cirkelbus, cisgender, citroenloopbaan, claimadvocaat, clickfarm, cloudclubbing, clownspiet, clubfunding, colorryn, comadrieken, comadrinker, comakijken, comazuipen, comazuiper, combikip, combinatiekip, combinatieklem, condoomplicht, conduct disorder, conflictdiamant, conflictgrondstof, consumptieprostitutie, contactberoep, contrastratio, cookiemuur, coolhanten, coolhunter, coolhunting, corona-app, coronahamsteren, coronakapsel, coronakilo, coronapas, corruptiewaakhond, cosplay, COVID, COVID-19, COVID-19-virus, covidoot, Creative Commonslicentie, creditboy, crimiclown, crisisdistillatie, cronut, crowdbutching, crowdfunden, crowdfunding, crowdfunding-site, crowdsourcing, crowdsuing, crowdtexting, cryomatie, cryomeren, cryptogeld, cryptomunt, cryptovaluta, curlingkind, curlingmoeder, curlingouder, customizen, cyberchondrie, cyberpester, daderinformatie, daderkennis, dagobertducktaks, dashcam, databundel, datahotel, dataretentie, dataroaming, datingapp, datingfraude, datingsite, declutteren, decolletéportemonnee, decryptiebevel, deelbakfiets, deeleconomie, deelstep, deeltijdvegetariër, deeplinken, dementievriendelijk, depodog, dertigersdip, designervagina, dessertkindje, detentieboot, deugmens, diabetes, dialoogtafel, dicteenomade, dieplinken, dierenpolitie, dieselgate, dieselhybride, dieselschandaal, digibesitas, DigiD, digipedo, digipesten, dijkloof, dikketruëndag, dislikebutton, divestment, divo, doe-het-zelfkassa, doellijncamera, doellijntechnologie,

domeinschool, donorpoep, doorverbinding, dopaminevasten, downdate, downdaten, downloadheffing, doxing, draagmoederbank, draaideurdocent, dramademocratie, drankcamera, drankhangen, drankkeet, dreigcultuur, driegeneratievakantie, drillrap, droneaanval, drugsplukker, dubsmash, dubsmashen, duckface, dumpsuper, duomoeder, duurzame hypotheek, e-auto, e-bakfiets, ebtl, ecocheque, ecoduiker, ecologische woongemeenschap, ecorexia, ecotainment, e-fiets, egoglossy, EHBO-kaart, EIOPA, elektrische brommobiel, emissieschandaal, emoji, emotiedemocratie, e-motor, emptynestsyndroom, energielogo, energiesnoeier, energievlieger, energiewaakhond, e-peritief, e-peritieven, epilepsiehond, e-rijder, ervaringseconomie, ervaringstuin, escaperoom, ESRB, etiketkind, eurowees, eurozondaar, e-vakantie, e-wagen, e-waste, fabeltjesfuij, facebooken, facebooker, Facebookvoyeur, facekini, facetimen, factchecken, familierechercheur, famping, fanboy, fanfiction, fangirl, fantoomgeluid, fantoomterreur, farm-to-table, fashionblog, fashionblogger, fashionista, fatkini, fatshaming, feestdagenstress, felfie, fembot, festivaluta, fietsblog, fietsblogger, fietsersairbag, fietsnietje, fijnstofflitspaal, filefuij, fileventiel, filterblokkade, filterbubbel, fitnessstracker, fittie, flexiseksueel, flexitariër, flexivoor, flexplek, flexvrijwilliger, flitshandel, flitshandelaar, flitshuwelijk, flitsinformatie, flitsmacht, flitsmenigte, flopkip, fluohesje, fomo, foodblog, foodbloggen, foodblogger, foodiebag, foodtrendwatcher, foodtruck, fotoblog, fotoblogger, fotobom, fotobommen, fotofuck, fotoshoppen, fotoweblog, free-floating, free-floatingstelsel, frexit, friendzone, frietchinees, frigoboxtoerisme, frosé, prosecco, Furbyfluisteraar, gameverslaafd, gameverslaving, gaslighting, gastblog, gastblogger, gaybrapad, gazondolk, geboortemoeder, gedoogakkoord, gedoogregering, geldezel, gemaksliteratuur, gemeentewiet, genderfluïde, genderfluïditeit, genderneutraal, genenspreekuur, generatiepact, generatieverzuijing, geolokalisering, gevoelsinflatie, ghosten, ghosting, gijzelsoftware, gijzelvirus, glampen, glamperen, glamping, gluurverhoging, goederenhotel, goedkopestoelenmaatschappij, goedmens, gojibes, goldendoodle, goodiebag, gorpcore, graaitaks, graftoerisme, gratispolitiek, greenwashing, grexit, griepremmer, grijsweger, groene cheque, groenpluk, groepsapp, groepsblog, groufie, gsm-parkeren, gsm-roman, gumcultuur, gutmensch, haatbaard, haatblog, haatblogger, haatimam, hackathon, halalhuis, halalhypotheek, halbeheffing, halsbandbom, hamburgerbelasting, hamburgertaks, handhavingsdag, hangbejaarde, hangoudere, hangry, hartfalenpoli, hartfalenpolikliniek, hashtag, hashtaggen, havercino, havermelkelite, healthie, heen-en-weerkind, heisessie, helikoptergeld, helikoptermama, helikoptermoeder, helikopterouder, helpersyndroom, herinneringskunst, h.jood, hipsterhoreca, hitteplan, hoestschaamte, holiday, hondenspeeltuin, honingsnelweg, horkenlijn, hospitawonen, hotelgezin, hotelpiraat, hoverboard, hufferhut, hufferindex, hufferwoning, huishoudentoeslag, huishoudwerkloosheid, huiskamerhuwelijk, huwelijksplanner, hybrideauto, hygge, hyperboek, hyperouder, ICE-nummer, identiteitsdiefstal, identiteitsfraude, ijsemmeruitdaging, IKEA-roman, incheckpaal, indigokind, influencer, infobesitas, infodemie, inhaalassistent, inklimmer, instagrammen, integritis, intelligente lockdown,

internetbellen, internetbubbel, internetdemocratie, internetdiploma, internetrechercheur, internetspaarrekening, internetsparen, internetsurveillant, internettrol, IPKO, IS-aanhanger, IS-lid, IS-olie, IS-strijder, jaarvak, jackpotting, jailbreak, jailbreaken, jarretelfilosofie, jegging, jeukwoord, jihadgezin, jihadi, jihadistrijder, jihadistteam, jinxen, jip-en-janneketaal, jokerjaar, JP-norm, jubelton, kaaspiet, kakuro, kalifaatganger, kalkspray, kamikazecoalitie, kamikazefiets, kanjertraining, kankerbos, kanswijk, kantoortijdenschool, kapperszaakconferentie, kattencafé, kendoementaliteit, kentekenparkeren, kernuitstap, keukentafelgesprek, keuzestress, keycord, keyloggen, keylogging, kibbelkabinet, kinderbrowser, kinderpardon, kindervrijheid, kindvrije, kledingbank, kledingbibliotheek, kledingcoach, kleptocratentaks, klets-kassa, klikfraude, klikhit, klimaatdrammer, klimaathypotheek, klimaatmigrant, klimaatneutraal, klimaatspijbelaar, klimaatspijbelen, klimaatstaken, klimaatvluchteling, klokkenluiderssite, klokkenluiderswebsite, knaldrang, knalpotterreureur, knieairbag, knieslot, knipooggeweld, knuffelcontact, knuffelhormoon, knuffelmagistraat, knutselcontract, kookblog, kookbloggen, kookblogger, koopkathedraal, koopvacuüm, kopvoetermaatschappij, kopvoetersamenleving, kraamkost, krachtwijk, kredietunie, kreukelpaal, krimp-dorp, kringwinkel, kristalkind, kruimelcontract, kuchschermer, kudos, kwaliteitstaxi, kweekburger, kweekvlees, laadinfrastructuur, laadpaalkleven, laadpaalklever, laadpas, laadstress, laagflatie, labradoedel, labradoodle, lagekostenluchtvaartmaatschappij, langstudeerdersboete, langstudeerdersmaatregel, laptopnomade, lawinerugzak, leefloner, leefloon, leefstijlcoach, lelijketruiendag, leukigheidsmaatschappij, levensloopregeling, levensloopstress, lgbt, lhbt, lhbti, lhbtqia, lichtwekker, lifestylecoach, likeknop, lippenstiffeminisme, locavor, locktail, loedermoeder, lokhipster, lokhoer, lokhomo, lokpuber, loktiener, lokwoning, lokzak, longread, longstayer, lonsdaler, loomarmbandje, loombandje, loopvoetbal, loungecafé, loungeclub, lovergirl, lowcostmaatschappij, luchtagent, luistercamera, luizenouder, luizenvader, luxespijbelen, maaiveldcultuur, MAC-adres, magneetpil, magneetziekenhuis, mancave, mangomoment, mansplaining, manspreading, mantelzorgboete, mantelzorgwoning, manty, Martin Garrixizer, medior, meditaxi, meelwormenpesto, meeroudergezin, meerwaardehypotheek, mekking, megxit, meisjesmeisje, menstruatiearmoede, metroman, metroseksueel, microgroente, mikadogezin, milieucheque, mindfulness, mindmap, mindmappen, minibibliotheek, minibieb, minibos, minister-presidentnorm, mipster, misofonie, misofoon, mobielverslaving, modeblog, modebloggen, modeblogger, modenomade, moestuinsocialisme, molloot, mooi-boy, moslimhypotheek, mozaïekgezin, mp3-spam, must-do, must-hear, muziekblog, muziekblogger, muziekweblog, naaktscan, naaktscanner, nahuwelijk, naming-and-shaming, natuurbrug, negen-tot-vijfmentaliteit, nekbom, neklint, neofundamentalist, nestklever, Netflix and chill, netflixen, netwerkgeneratie, neuromarketing, nexit, niesschaamte, nieuwsblog, NIFP, no-gogebied, no-gozone, nomofobie, normcore, notaristoerisme, NRGD, nultolerantiebeleid, Obamacare, omkeerhypotheek, omkeertijd, onderwaterhypotheek, onesie, onroerendgoedbubbel,

ontdierentuinen, onthoofdingsvideo, ontrommelen, ontpullen, ontvaderen, ontvadering, ontvolgen, ontvrienden, ontwijnen, ontzooën, openbronsoftware, opensourcesoftware, ophokuur, oplaadinfrastructuur, oplaadpaal, oplaadpas, oppimpen, oprotzes, opslagval, opvoedgezin, opvoedpoli, oranje-groene stroom, orthorexia, overgewichtbelasting, overlastboete, overlastcel, overlastdonatie, overlasttaks, overschotdoos, ov-fiets, pafpaal, pakjesbus, pakkenproletariaat, pampergeneratie, pamperrekening, pandapunt, pandemoe, papadag, Paraplurevolutie, passiefbouw, passiefflat, passiefhuis, passiefkantoor, passiefnorm, passiefschool, passiefwijk, passiefwoning, pastafarian, pastafarianisme, pastafarisme, pastaprinter, patchworkgezin, patiënttevredenheid, paywall, pdf-spam, pechakucha, pechstrookrijder, pedopriester, persieing, peukenput, peukentegel, PFAS, phablet, phubben, phubbing, piekorgasme, pikettysme, pimpen, pinkassa, pinsparen, plaknacht, planker, planking, plassticker, plasticsoep, plastiglomeraat, plavriendelijk, plaswinkel, pleegverlof, pleegzorgverlof, plofkip, plofklas, plofzak, plug-inhybride, plusmama, plusmoeder, plusouder, pluspapa, plusvader, podcast, podcasten, podcaster, podcasting, poederbrief, poepbacterie, poepbank, poeptransplantatie, polderboeddisme, poldercatenaccio, polderhunk, polderjihadist, poldermodderen, poldersafari, policor, politainment, politierat, pooierboy, poortjesspringer, pop-upmuseum, pop-uprestaurant, pop-upshop, pop-upstore, pop-upvenster, pop-upwinkel, porna, pornoknop, postcoderacisme, postcodezorg, potel, powernap, preferendum, premiernorm, preteaching, pretparkgeneratie, prexit, productplaatsing, promotieval, prullenbakvaccin, psychiatriebijbel, pullka, pushbericht, pyroshow, QR-code, quadrikooper, quarantainer, quarantinderen, quarterlifecrisis, quotilde, raamambtenaar, raamvisite, racismeregister, ramadanpil, ransomware, reaguurder, recessionista, reclameluw, regenboogkind, regenboogzebra, regenboogzebrapad, regiearts, regiedokter, regieverpleegkundige, reisblog, reisblogger, relikliniek, reltoerist, repaircafé, restorestje, retweet, retweeten, robothotel, robottruck, robotvalk, roetpiet, rokersabri, rokershut, rokjesdag, rolezinho, rollende keuken, romcom, rommelhypotheek, rommelstatus, roodkapjesyndroom, rookhut, rookpaal, rouwcoach, rugzakbeleid, rugzakfinanciering, rugzakgeld, rugzakleerling, rugzakwet, ruilseks, salariswagen, Salduzwet, salonpopulisme, salonpopulist, sapioseksueel, SARS-patiënt, schaduwelite, scharrelboom, scharrelkind, scheefhuur, scheefhuurder, scheefwerken, scheefwoonboete, schermtijd, schoolbelbaan, schoolwijkagent, schoorsteenpiet, schoudersurfen, schreeuwcultuur, schuurei, scoxit, screenager, scroppino, scrummen, securityscan, securityscanner, segway, seizoensgriep, selfie, selfiestick, selfiestok, selfitis, seniorenmoment, seniorenstad, serieverslaving, sexercisen, sexting, sextortion, shelfie, sheltersuit, shishabar, shishacafé, shishalounge, shishapen, shockblog, shufflegeneratie, singleschaamte, sjoemelsoftware, skaeve huse, skinny jeans, slaaprijden, slaaprijder, slashie, sletvrees, sleutelgatchirurg, sleutelkoord, slikschaamte, sliplift, slodderwetenschap, slowfoodbeweging, slurptaks, slushie, slutshaming, smartbed, smartcamera, smartphoneverslaving, smirten, smombie, sms-

parkeren, snapchatten, sneakpeek, snelheidsdeken, snellaadpaal, snellaadpunt, snellaadstation, snoepfruit, snoepgroente, snotterschaamte, snuifschaamte, social distancing, social media, socialbesitas, sociale media, sociale onthouding, sofapatat, sonjabakkeren, spambericht, spamblog, spamfilter, spammail, spandoekautomaat, speedbike, speeddate, speeddaten, speedpedelec, spelletjesverslaafd, spermahond, spiegelcoalitie, spiegelregering, spitsalarm, splashteller, splinterkwartier, spoedzoeker, spookburger, spookforens, spookjongere, spoorboekloos, spornoseksueel, spouwdonut, spuugkit, spuugmasker, spyware, stadsblog, stadsblogger, stadskip, stadsstrand, staleend, stamcelbank, statusupdate, steekvest, steekvlampolitiek, stekkergeld, stekkerhybride, stekkerwagen, stemfie, stempas, sterke AI, sterrenkind, stiltezone, stoeproken, stoptober, straalkanon, straatintimidatie, streetfishing, strooifiets, stroopwafelpiet, struikelmat, studiekeuzecheck, stuurairbag, subprime, succesgezin, sudoku, sudokuen, suikertaks, superfood, suppen, swaffelaar, swaffelen, swipen, synergiebonus, Syriëganger, taakomroep, taalmaatje, taalnazi, tabletbochel, tabletnek, talencafé, talibaniseren, talibanisering, talkshowelite, taseren, tatta, tattoogate, taugate, teamsen, teenschoen, telefoonverslaving, tentsletje, tepelgate, tepelsticker, terroroehoe, terugpinnen, testsamenleving, themablog, thuisblijfmoeder, thuisblijfvader, thuiestappen, thuiestapsysteem, tiktokken, tilgrens, tinderellasyndroom, tinderen, tiny forest, tiny house, toiletwinkel, tomtomburger, toogviroloog, topdorp, Tozo, trabisafari, transferunie, traumabus, tregging, trekteller, trending topic, treurtrip, triobaan, trollenleger, trollennetwerk, trouwplanner, T-shirtbeha, tuigdorp, tuinblogger, tweedelegvader, tweep, tweet, tweeten, tweevingertest, twerken, twexit, twitteraar, twitteraarster, Twitteractiviteit, twitteratuur, twitteren, Twitterspreekuur, überseksueel, uitcheckpaal, uitgestelde friet, uitstelbaby, uitstelvader, uitvaarthotel, uitvaarttoerisme, uitzichtgarantie, ultracultuur, ultragroep, UMTS-veiling, urilift, vegetariër, valpoli, vatbom, velostrade, vergeetverzoek, vergisliquidatie, vergismoord, vergisontvoering, verkeersalarm, verklikkerkip, verlofcultuur, verteltas, vertiktokking, verwezen, verzetsspray, Veteranenloket, vettaks, videoblog, videoblogger, videoselfie, villabelasting, vind-ik-leuk, vind-ik-leukknop, vind-ik-niet-leukknop, vingervoedsel, visdeurbel, vismigratierivier, vitaliteitsregeling, vivaldicoalitie, vlaggenschipwinkel, vleesminderaar, vleesschaamte, vlexit, vliegschaamte, vlinderakkoord, vlinderregering, vlij, vlog, vloggen, vlogger, vlogster, vluchtbewaker, vluchtstrookrijder, voedingscoach, voedselblog, voedselblogger, voedseltrendwatcher, vogelpindakaas, voipen, volg-me-nietregister, vondelingenluik, vondelingenschuif, voorschotfraude, voorsteekpas, vorst-WW, vrachtpatser, vrachtram, vrije-uitloopkip, vroempartij, vroemvroempartij, vrouwentaxi, vuilbakschool, wachtverzachter, Wajonger, Wajongere, Wajong-gerechtigd, Wajong-gerechtigde, wallah, wangiri, wappie, warmetriendag, warmtezuil, watergel, waterpijpbare, waterproductiviteit, waterwappie, watskeburt, webinar, webloggen, weblogger, weeralarm, weesfiets, weeshuistoerisme, wefie, wegjorissen, wegkijkstaat, weigerambtenaar, weigerarts, weigerburgemeester, weigerondernemer, weigerpastoor, weigervrij, welkomwinkel,

welzijnscoach, wenswachtende, werkfruit, werkverdringing, whatsapp, WhatsAppbericht, whatsappen, WhatsAppgroep, widget, wifi, wifitracking, wiën, wijkblog, wijnblog, wiki, wildbreien, wilfen, wilfer, winkelsafari, wittejasseneffect, woekerpensioen, woekerpolis, wokakkoord, wokrestaurant, wollah, woonuniversiteit, woon-zorgzone, wormenburger, wraakporno, wtf, wuifmoment, wurgnet, WW-toerisme, yammeren, yogajes, yogasnuiver, yolo, youtuben, youtuber, yuccie, zakgeldfederalisme, zandkasteleffect, zebra-klok, zedenadvocaat, zedenhond, zedenverdachte, zeeboer, zeilensnijder, zelfdodingsconsulent, zelfroosteren, zelfscankassa, zelfspiritualiteit, zerotoleranceaanpak, zesjescultuur, zeurkassa, zeurterreur, zijdecolleté, zikini, zilveren goal, zkv, zoekbot, zoekmachineoptimalisatie, zombiecomputer, zombienetwerk, zomerlek, zondagshoppen, Zoneparc, zooikoorts, zoomzombie, zorgautoriteit, zorggevangenis, zorghorloge, zorgidentificatienummer, zorgnummer, zorgpark, zorgplek, zorgpremie, zorgtoerisme, zorgtoerist, zorgtoeslag, zorgwerker, zoutkaartje, zoutloket, zuipbus, zuipkeet, zuiptoerisme, zumba, zwakke AI, zwartekoppenziekte, zwerfkapitalist, zwerflawaai, zwerfzwangere, zwijgprotest, zwijgshikking

Niet behouden neologismen uit het WNW (N=1104)

1,5 metermaatschappij, 1 literauto, 1 literwagen, 100 procentscontrole, 11 september-effect, 24 uursschool, 2G, 45 kilometerauto, 45km-auto, 45km-wagen, 60-minmaatschappij, 65 plushypotheek, aanbiddingszanger, aanbodgedreven, aandachtsminuut, aanmailbaar, aanraakgevoelig, aardgastankinstallatie, abb, abortustoerist, abri, acceptatiestop, accordeonhypotheek, accuangst, accustress, achterbanweek, achterhoedeschool, achterstandsscholier, activeringstaks, Actualisering Vinex, ADHD-klas, adob, adopteerkerstboom, adoptiekip, adviesethiek, aed, afdekcultuur, afhaalfile, AFM, afrekencultuur, afromkassa, afstandsmaatschappij, afstandssamenleving, afstudeerhotel, aftelstoplicht, afterclub, afvaltak, afwijsfabriek, agile, agressieharnas, agressiespoorlijn, agrotourist, aidsambassadeur, airbaghelm, Airbnb-economie, aka, alcoholblokkade, alcoholofiel, alcoholpoeder, alleenpinnenkassa, alszaak, ambulancekleven, ambutaxi, anderhalfgeneratie, anderhalvemeter-, anderhalvemetercoach, anderhalvemetersyndroom, anonieme sjaal, antiaanbakfolie, anti-allergiekoek, antibioticateteam, antibladerlaser, antiblarenspray, anticonceptiecheque, antigraaibeding, antikopieersoftware, antipiraterijsoftware, antischoonochterclausule, antischoonzoonclausule, antispamsite, antispamwebsite, antispijbelchip, antispuugkap, antistoller, antiverdwaalarmband, antiverdwaalpaal, antraxbrief, AOV, AOW-toerist, apartheidsbus, apenstaartgeneratie, app, armoedemigrant, artjockey, asobestendig, aspergeoogstmachine, aspergesteekmachine, atoomwaakhond, audiobloggen, audioblogger, auditwaakhond, autokloner, autostamboom, AVG, avontuurgen, awkward, babe, babyblog, babyblogger, babybus, babycrash, babymoonvakantie, bacterieafdruk, badstanding, baguetting, bakfietsen, bakfiets-sale, balanceboard, balansburger, balietoeslag, ballenhond, balletjespistool, bandenhotel, banensite, bankshop, banksparen, banktaks, barebackfeest, barebackparty, basejump, bashen, bashing,

batterijlaadstation, bbb, beachsoccertoernooi, beamer, beatbloggen, beatblogger, beatblogging, beautytoerist, bed, bad en brood, bedgebondenheid, bedrijfswagenschaamte, bedtaks, beef, beeldbellen, beeldborrel, beeldborrelen, beeldsnack, beleveniseconomie, belevingscultuur, belevingsonderzoek, belhuis, belplafond, belwinkel, bengelbus, bermbomaanval, beroepsmoslim, BES-bestuurder, beschermingsmuur, beschermingswal, betaaltechnologie, bewonersparkeren, bff, BGL, biblioblog, biblioblogger, bicultureel, biechtwebsite, bierviltjesviroloog, bigbrotherapp, bijenhalte, bil-bh, bioaanval, biobedrijf, biolandbouw, biometrisch paspoort, biopas, biopaspoort, biopiraterij, bioraffinage, biotech, bioterreur, bioverkoop, biowaterstof, blingbling, bloedpaspoort, bloedweigerpasje, bloembollenasiel, bloghoppen, bloghopper, blokketen, blootscan, blowgebodsbord, blowzonebord, bluetoothtracking, blurren, bobotaks, bodemei, bodypacker, bodyscan, bodyscanner, boefproof, boekenbank, boekenpoef, boetepolis, bofkip, bokwa, bollenasiel, bombox, bomdatabank, bomdatabase, bonding, bonusrenner, boodschappenbelasting, boombrug, boomer, botoxtandarts, bouwvakkerslook, boycottomaat, breedparkeerder, breedparkeren, broekjesruil, broodnood, bruiloftsplanner, brulouder, bsn, bubbel, buddymoon, budgethuwelijk, budgetsupermarkt, bugnugget, bugsballetje, buidelwoning, bumpwatching, burgerservicenummer, busboot, bushokje, bussafari, buttkick, buurtpreventist, buurtregisseur, cabriodak, camera-gsm, cameramobieltje, cameratelefoon, camjo, cammobieltje, camperen, cappuccinoloopbaan, cardtrapping, carrièrepil, cartoongate, cartoonkrant, CBP, celbie, censuurploeg, chill, churchie, CIOT, cloud, clutch, Cn, cobot, cockerpoe, cocoonvaccinatie, coderoodinflatie, colataks, cold case, coldcaseteam, collecteapp, colorblocken, combinatietestament, combitaks, compensatievakantie, computerbank, computerspelersslaafd, condoomboom, connectietarief, constrictor, consumentloket, contactloos betalen, containerkruiper, containertuintje, corona, coronaratel, cosmeticaverslaving, cougar, cougarcruise, covidium, crisisheffing, cyberaanval, cyberbabe, cyberchonder, cybercrime, cybercriminaliteit, cyberinsect, cyberinspecteur, cybermisdaad, cyberoorlog, cyberpesten, cyberpolitie, cyberterreur, cyberterrorisme, cyberterrorist, dabben, daderwetenschap, daten, daterapedrug, datingfraudeur, daytrader, deathcam, deelfiets, denimcyclus, desertpak, designbaby, designdrug, deugschaamte, deurwijzer, dikkentaks, dinsdagdilemma, diplomabank, disco-oor, dislikeknop, doe-economie, doekoe, doellijnbewaking, doe prikkel, dokterbibberregel, dolfijnsafari, doodle, doorstarter, dorpsmanager, dorstsensor, doula, downleren, downloadbelasting, downloadtaks, draaideurcrimineel, dresskini, drieouder-ivf, drieouderschap, drinkpiraterij, drolemoji, droneclub, droogijsbom, dubbellezer, duimgeneratie, dumpkind, dumpsupermarkt, duurzaamheidsambtenaar, dweil, dweilen, earlyaccessprogramma, earlyaccessstudie, EBA, e-bike, e-boek, ebolabel, e-book, e-bromfiets, e-brommer, e-brommobiel, ecofobie, ecoviaduct, e-deelbakfiets, edublog, edubloggen, edublogger, eenhoorn, eerste-uitgiftegesprek, eitjesinvriesparty, elektrische 45 kilometerauto, elektrische 45km-auto, eliminatiebank, emocratie, emokapsel, e-motorfiets, e-motorscooter, end-to-endbeveiliging, energieneutraal, energietransitie, energie-uit-afvalcentrale, e-rechtbank, ervaringsbewijs, ervaringsmarketing, e-scooter, ESMA, e-snorfiets, e-step, e-toxen, E-woord, exfiltrant, fail, fair trade, fairfokhond, fairfokkeurmerk, fakeboy, fashionbloggen, fashiorexia, fecesbank, femiman, festivaladvocaat, fietsairbag, fietsbellen, filesluiper, fingerfood, flashmob, fleet, flexfuelwagen, flexwerkershypotheek, flipperen, flipperkastcommunicatie, fluisterbehang, fluistersolden, foodie, fopduik, fotofucken,

frame, framen, framing, free-floatingmodel, fusionrestaurant, fygitaal, garagemoskee, gasrotonde, gastbloggen, gaybra, geboortenaam, geeninkomenhypotheek, Geldergate, geluidshaat, gendervakantie, generatie X, generatie Y, generatiehypotheek, gespreksnarcisme, geuralarm, geurbank, gezelligheidsracisme, gezichtsbikini, gezichtstransplantatie, gezinscoach, gieg, gifbibliotheek, gijzelmalware, girlpower, glazenbolsubsidie, goededoelenmerk, goldendoedel, gordijnairbag, gordijnkiezer, gordijnstemmer, gps-schoen, graaispaarder, gramping, granaatwurger, grasfalt, groentenugget, groenvrijwilliger, groepsbloggen, groepsblogger, guerrillaschommel, guerrillaschommelaar, guerrillaschommelen, halaliseren, halsbom, hamburgermenu, hangburger, hangkind, hangklant, hangsenior, hartnavigator, harttomtom, hasjpil, heartbeat, heidesafari, heligate, helikopterkind, helpie, herdenkingscollage, hersengadget, herstelrecht, hesje, heupairbag, himbo, hipster, hobbyen, hockeytracker, hoestdab, homejacker, hommelsnelweg, homocide, homowijk, hotel mama, hotspot, hubby, hufferbunker, hufferhuis, huisdierenpolitie, hybridiediesel, hyperopvoeder, ICBC, icd, i-generatie, ijskastdiplomatie, impulsklas, infographic, infra, inlognaam, inruilboom, insectensnack, internaut, internetbankieren, internetcriminaliteit, internetfiets, internetgedrag, internetinspecteur, internetlek, internetondernemer, internetparkeren, internetrechtbank, internetspaarder, internettijd, intimidatiecultuur, IS, IS-benzine, jagger, jihadist, jihadofficier, jihadvakantie, jomo, junkfoodbelasting, junkfoodtaks, juwelier, kaasnugget, kalifaatvrij, kamelenchocolade, kamelenhoest, kamelenhoestpatiënt, kamelenkuch, kamelenweg, kamikazesatelliet, kantelkenteken, kantoorurenschool, kantoorwandelen, kantoorwoning, kapsalon, keylogger, kijkcijferkanon, kiloknaller, kindervrij, kindervrije, kindlint, kindvrij, klikocrimineel, klimaatwaakhond, klokhuisboom, knakaal, knuffelmissie, knuffelpraktijk, koekoekwonen, koopjessafari, koopknop, kopschopkaker, kot couture, kraagbom, krokodil, kuchsot, kufiyyah, kuroburger, kwekplek, laadinfra, laadstation, laatstekansvader, laatstekanswoning, ladette, lanyard, latrinevrees, lattepapa, lawineairbag, leeftijdsmunt, leenangst, leenkerstboom, leesblog, legotelefoon, levensmiddelenwaakhond, lgbt'er, lgbtqiap, lhbt'er, lhbti'er, lifeblog, lifebloggen, lifeblogger, lifehack, ligrestaurant, lijger, like, liken, linkblog, linkblogger, lipotoerist, lockdown, locktalgie, logeervergoeding, lok-bh, lokcomputer, lokkoffer, lokschaap, lokstudent, lonewolftheorie, loods, loomelastiekje, loomen, loungen, loveseat, lowfaremaatschappij, L-teken, luchtkavel, luismoeder, luisouder, luisvader, luxespijbelaar, mabelen, magneetzeep, magnieter, mammapolikliniek, manny, manteluur, mantyhose, mededingingswaakhond, medicomaat, meerkatten, melk-suikerklontje, meme, menstruatievrij, mental coach, mention, mijdwoord, mikadogeneratie, milf, milieudrammer, millennial, millenniumgeneratie, miltvuurbrief, mindfuck, misofonielijder, mobiele aanbieder, mobielelefoonverslaving, moedermaffia, moedermelkijs, moedertheresasyndroom, moerassafari, monnikplek, monokini, moonen, moshpit, mosquito, movel, mup, muppie, must-have, must-read, must-see, muziekbloggen, myocardaantal, mysteryshoppen, nachtgat, nagelsticker, nanotijdperk, narcomobilist, narcosebox, NCTb, NCTV, nederspeed, nefrostick, nekairbag, neknominaten, neknominating, neknomineren, neoyup, nepbaasfraude, nepfietsstrook, nepnieuws, netflixisering, netwerkgezin, nieuwsbloggen, nieuwsblogger, nieuwsrat, nocturnivoor, nokay, nordic walking, NOW, nul-op-de-meterbouw, nul-op-de-meterbouwen, nul-op-de-meterhuis, nul-op-de-meterhypotheek, nul-op-de-meternieuwbouw, nul-op-de-meterrenovatie, nuloudergezin, nultolerantieaanpak, obesitaks, obesitasbelasting, oldskool, omfietswijn, OMG, onderkoningin, onlinemoord, ontdopen, ontdoping,

oogbaltatoeage, oogbaltattoo, opblaaskerk, opdringkassa, ophokken, ophokplicht, opkotten, oplaadinfra, oplaadstation, oppoppen, opvoedouder, oranjetompoesindex, out of the box, outside the box, overlastbelasting, overlastmanager, overlastteam, over-the-countergeneesmiddel, over-the-countermedicijn, paalklever, paasei, pad, pafdak, pafpremie, pakketdrone, pakketstation, pakkettenbus, pakkettenkast, paniekknop, panna, partenariaat, participatiemaatschappij, participatiesamenleving, partycrashen, partydrug, passiefadministratiegebouw, passiefbouwwooning, passiefkantoorgebouw, pauzewoning, pedelec, peerfie, peilingen zijn palingen, pelotonremmer, peukenplukker, piekdrinken, pieperschieter, piggate, pindarotsman, pineberry, pingpongcommunicatie, pinkwashing, pin-onlykassa, pinpendel, planken, plastuit, pleegkerstboom, pleisterambtenaar, pleisterpil, plexiglassamenleving, plug-in, plug-inhybrideauto, plug-inhybridevoertuig, plug-inhybridewagen, poco, polder, politiehypotheek, postcodediscriminatie, prachtwijk, preformateur, prikvertikker, privacywaakhond, propper, PUB-ploeg, puntjesboek, pyjamadag, pyjamawerker, queer, rapedrug, raw food, reallifesoap, recessiegroeier, recyclekerstboom, redderbaby, regensensor, retourkerstboom, retourpinnen, risicohypotheek, roamen, roaming, rokersafdak, rokersaquarium, rokerskooi, rokerspremie, rollercoastercursus, romkom, rommelblind, rommelblindheid, rookairbag, rookaquarium, rookkooi, rookzuil, rouwclown, rugzakindicatie, rugzakker, saldiniest, saldosjoemelaar, salonviroloog, samenzwever, samenzweving, sapreinigingsvakantie, SARS, SARS-virus, sauzen, scamtelevisie, scharniermagistraat, scheefhuren, scheefwerker, schijnstudent, schoolfruit, schoolstraat, schuimecho, scootmobiel, scorebordjournalistiek, screenshot, seizurehond, seksihadisme, sellofie, semestervak, seniorenfile, sensatiespiraal, SEO, sharent, shoulderen, s-icd, sigarendoosviroloog, sigarettenhoesje, sigarettenindex, silver goal, sintenschisma, skegging, skifile, skinny, skypen, slaapwachtende, slechtweergarantie, slimwoner, slofie, slowfood, slowfoodgerecht, slush, slushpuppie®, smaakjестhee, smartpen, smelt, smoothie, snackbelasting, snackmoment, snackseks, snacktaks, snelfietsroute, snelrechtbus, snelwegbus, snikkelgoal, snorkelgen, snorkelrijst, socialemediatestament, sojacino, sollicitatieloon, sologezin, spider, spijbelchip, spinazienugget, spitsmijder, spitsuurgezin, spooketage, spookfile, spookinwoner, spookstudent, spookuitkering, spraycondoom, spudgun, spuugkap, SRD, stadsbloggen, stadsmarinier, stapspion, startershypotheek, stationbased, stawerkplek, sta-zitwerkplek, steedsmeergeneratie, straatbijbel, straatsen, strategisch akkoord, streetfood, stressorexia, stressziekenhuis, subprimehypotheek, sueën, suïcidetoerisme, suikermaxdag, surfminuut, susser, susteam, suswacht, suswachter, swa, swag, swagger, swap, swexit, swingerscruise, taarten, tabletkramp, tandpastabom, tegelasbak, telefoonallergie, terf, terrasheater, terreurbenzine, terreurbroer, terreuroehoe, textielantenne, themabloggen, themablogger, thuishapper, thuisenaar, tijgermoeder, tijgerpunt, tikfoutdomein, time-outgezin, tjoep, to go, toiletangst, tomatenpil, TONK, trappelkaart, travoltariseren, treinsafari, trending, trilsjaal, tsunami, tuinblog, tuinbloggen, tunnel, tv-babe, TVL, twandeling, tweedeverblijver, tweenager, tweeter, tweetshop, twexcursie, twibbatical, Twitterexcursie, twitterviroloog, Twitterwandeling, twittie, typefoutdomein, überopvoeder, überouder, uilencafé, uitpakvideo, uitvaarttoerist, unicorn, upskirting, Uus, Uut, vaccinatievoordringer, vaccinatievoorsteker, valleiorgasme, valpolikliniek, valsimulator, vastgoedbubbel, vastgoedzeepbel, veejay, veggie, verbodsethiek, verdwaalarmband, verdwijnsubsidie, verhuftering, verkeerdebedpatiënt, verklikkip, verkrachtingsdrug, verlengde huiskamer, vermeldingswaarde, versoepelstress,

vertalbaniseren, vertiktokken, vetbelasting, viben, videojockey, virtuvrijen, virusschrijver, vitalo, vj, vlakkevloertheater, vliegtol, vliegtuigcruise, vliedervagina, voedselbloggen, vogelflap, volglocatie, vook, vrije-uitloopouder, vrijgavemunt, vrijlandkip, vrijmibo, vruchtbaarheidstoerisme, Wabo, walkingfootball, wandelvoetbal, wangirifraude, wannahave, wasmisser, waterkip, waterstofauto, wearable, webpesten, weermelding, wegwijnziekte, weidesafari, weigerparlementariër, wellnesscentrum, wenswoning, werknemersarts, WiDi, wijkbloggen, wijkblogger, wijk-ggz'er, wijkschool, wijnbloggen, wijnblogger, wijnen, wijnslushie, wildbraken, wildroker, wildschommel, wildschommelaar, wildschommelen, witte school, witteroekmoment, woela, woke, woodcraft, woonblog, woonbloggen, woonblogger, woonschaamte, woon-zorgappartement, woon-zorgpark, WOZ-schade, xenolavabofobie, xenosboeddhisme, xenotransplantatie, yammeraar, yo-app, YouTubegebruiker, yuc, zaadkind, zebraadrempele, zeemolen, zelfbecijfering, zelfdodingsconsulente, zelfmoordconsulent, zelfmoordconsulente, zelfstandigenhypotheek, zeurvrij, zezel, ziekte-cv, zijlijnviroloog, zilveren treffer, zit-stawerkplek, zitstep, zitwerkplek, zoekverkeer, zombie, zone-eigen, zonneraceauto, zony, zoomen, zorben, zorgfraude, zorghulp, zwarte school, zwavelarme diesel, zwembadprotocol, zwemverbod, zwerfbus, zwerffiets, zwexit, zwijgdemonstratie, zwijnneusrat, zzp'en

Appendix II: Neologismen uit *Neoloog*

Behouden neologismen uit Neoloog (N=2293)

0-procentbijtellingsregeling, 1-euromaaltijden, 15-dollarbeweging, 2-gradendoelstelling, 2020-strategie, 24-uursnieuwszender, 24-uursvoorstelling, 3x3-basketbal, 3x3-basketbalploeg, 450-programma, 60-procentmaatregel, 60-procentverplichting, aandachtsspons, aankomstcultuur, aanleunappartementen, aardappelpizza, aardbevingsgeneratie, aasgierfondsinitiatief, accordeonformule, accordeonkrediet, achterdeschermenfilm, actuatandem, actuasong, aderpatroonherkenning, adoptiemanager, adoptiemeter, advertentiebelasting, afbakeningsdecreet, afhaalluik, afhaalmoment, afluistercomplex, afrekenpaal, afrekentoets, aftoetsronde, afvalwaterinjectie, afwikkelingsfonds, agendaapp, agreekment, agressieset, agroafval, agroecologie, agroverzamelgebouw, alcoholbril, alcoholhetze, alcoholresolutie, alcoholslotmaatregel, alcoholslotprogramma, alcoholstrategie, alcolab, alfafunctie, alibiallochtone, allyoucaneat, allemansbank, allesuitdekastscenario, ambassadeurssysteem, ambelopoulia, amberalert, ambtenarisering, angstprediker, ankerbabys, anprcameras, antiaardbevingsstructuur, antibierfietspetitie, antibusiness, anticorruptieexpert, antieuropartij, antieuropartij, antiglobalisten, antigraffitibekleding, antigrowshopwet, antiimmigrantprogramma, antiimmigratiebeleid, antiimmigratiemars, antiimmigratiestandpunten, antiislambeweging, antiislamdemonstratie, antiislamisering, antiislamiseringsbeweging, antiislammeeting, antiislamofobie, antiislamfoob, antiislamplan, antiislamretoriek, antiislamvereniging, antijihadbijeenkomsten, antilogoclausule, antimelklobby, antimigrantendiscours, antimoslimbetoging, antimoslimdiscours, antimoslimplan, antipiratenoperatie, antishiitische, antistarchitect, antiterreuralaam, antiterreurmaatregelen, antiterreurorgaan, antiterrorismecoördinator, antiterrorismemaatregelen, antivaccinatiebeweging, antizwartepietbeweging, antibotox, anticorruptiewaakhond, anticorruptieaanklagers, antimelklobby, antimigratiediscours, antimigratiestemming, antimisbruikwet, antiradicaliseringsambtenaar, antiradicaliseringscel, antireligiestemming, antiterreuragenten, antiterreurrechters, antiterreurcoördinator, antiterrorismeoperatie, antiterrorismespecialist, antiterreuroperatie, antiwitwasdienst, appbank, appbedrijfjes, appberichten, appbouwers, appcultuur, appeconomie, appexpertise, appgedrag, appgroep, apphulp, appideologie, appmakers, appontwikkelaar, apprevolutie, appscène, appstore, appstreamen, appversie, appwinkels, appdiensten, appnek, arbitragemechanisme, architectenbalie, architectuurapp, architectuurglossys, armoedeindicatoren, armoedemeldpunt, armoesaus, arthousecinema, arthousefilms, artiestenloft, artspeak, asbestcementregister, asielcodex, asieldoorgang, asieldorfpen, asielinterviews, asielzoekas, asielzoekerij, attentiejunk, audiouitgever, autituin, autogps, autooplaadpunt, autoreply, autodeelapp, autodeelplaats, autodeelplan, autodeelplatformen, autodeelstad, autodeelverzekering, autodeleninitiatief,

automijdend, autoplayfunctie, avantgardekeuken, avondplaylist, babbelauto, babekleren, babyccchino, bachelornabachelor, badeendkomeet, bailin, bailoutplan, bailoutregeling, bakfietsparking, bakfietswijken, baksteenindemaaggevoel, balansissue, ballonnenactivist, ballsy, bambigevoel, bankingapp, basisapps, basisbereikbaarheid, basiswoonbonus, bayakou, bbgun, bc/mcapp, bdsmfilm, beauf, beautyvlogger, bedbad, bedbadbroodopvang, bedbadbroodregeling, bedbadbroodconflict, bedbadbroodcontroversie, bedbadbroodcrisis, bedbadbrooddiscussie, bedbadbroodlocaties, bedbadenbrood, bedbadenbrooddiscussie, bedbadenbroodopvang, bedbadenbroodcrisis, bedbadenbroodstandpunt, bedelnota, beeldmarketing, beeldrecyclage, beeldtijd, begrafenisjaar, begrotingsboekhouden, beiaardapp, bekeringsbehandeling, bekeringskantoor, bekeringstherapieën, belastinglingo, beleefdorp, beleggingscomputer, beleggingstrainer, beleidsbubbel, belevenispark, belevenisstad, belevenistrend, belevingsaanpak, belevingscentrum, belevingsdag, belevingsinstallatie, belevingssupermarkt, belevingstrap, belevingstrend, belgezellen, belgotariër, belgotariërsdieet, belhistoriek, beloningscommissaris, benefiethouding, benoemingsingreep, berekeningstool, berendruifextract, berichtenapps, beroepstreiteraar, beschadigingsoffensief, beschermlandschap, beslishulpen, besparingswatch, bestemmingslus, betaalapp, betaaldeer, betaalplein, betaalsportzender, betaaltoestel, betonneritis, beurspauze, beveiligingscookies, bevingskamer, bewakingscultuur, beweegonderzoeker, bewegingscoördinator, bewonersidee, bezettingslening, bezorgdrone, biebezel, biefstukkentijdperk, bierfietsondernemers, bierfietsterreur, bierpiraten, bietenburgers, bigdatabedrijfje, bijnaenergieneutraal, bijspijkermarkt, bikinibabes, bingedinking, bingeinterviewen, bingewaardige, bingewatching, bingekijker, bingekijkt, binglezen, bingemens, binnenlandsafari, biogecertificeerd, bioinformatici, biolimonades, bioondergoed, bioautomaat, biodrama, biohalalwinkel, biohoeve, biosimilar, biosimilarfabrikant, biosmoothies, bitcoinadepten, bitcoinadres, bitcoinblockchain, bitcoincommunity, bitcoinevangelie, bitcoingemeenschap, bitcoinhype, bitcoinland, bitcoinnetwerk, bitcoinbedenker, bitcoinfilmpje, bitcoinlogo, bitrot, blackfaceact, blacklivesmatter, blingblinggeneratie, blitzkriegstijl, blockbustertarief, blockchainonderzoeker, bloggingsite, blokcentrum, blokspot, bluebike, bluetoothinterferentie, bluetoothmeting, bluetoothring, blusbusje, boekenapp, boekenbrunch, boekingsapp, boerenkoolchipje, boerenkoolpesto, boerenkoolsmoothies, boerenkoolwedstrijd, boerkiniverbod, boetiekfestival, boksmode, bonnetjesrapport, bonusmogelijkheid, bonusnorm, boogiegewijs, boogschuttergebaar, borstvoedingskussens, borstvoedingspecialzaak, borstvoedingsthee, bottomupmarkt, bouwmarathon, breakevendrempel, breakupliedjes, breakupsong, breincrisis, breinprojecten, breuktaaiheid, breuktaaiheidstesten, brievenbuskantoor, brilhulp, broccolicouscous, brontosaurussteak, broodbedbad, broodvoorzieningen, broodcoupon, browserextensie, brunchzaak, buckybom, buddyhond, budgetarmoede, budgetpolissen, budgettaxidienst, buffelroomijs, bufferstock, buisconcert,

buitenlandertol, buitenlandsaccentsyndroom, bunq, bunqapp, burgerdienstproject, burgerlabo, burgermeldpunten, burgermobilisatie, burrneshas, businesstocustomercommunicatie, busjeopbusjeaf, buurderijsysteem, buurtengagement, buurtontmoetingsplek, buurtstudios, buurtzorgcollectief, cabaretcollege, cafeïnepil, caipirinha, calabaza, calimerogedrag, cameradrone, cameratraject, camperclub, caremodel, cargohub, carnavalsbussen, carrièreboost, carrière syndroom, cartoondiplomatie, cashloos, cashplafond, cfd, chatapp, chatcorpus, chatgenre, chatgeschiedenis, chatgroepjes, chattaal fenomeen, chichimadammen, chicklitromans, chilloutruimte, chilloutzone, chillsfeertje, chipszakeffect, chocoladetherapie, cholera pak, Citroentijmboer, claimorganisatie, clipperformance, cloudindustrie, cloudleveranciers, cloudopslagdienst, cloudrobotica, cloudserver, clouddivisie, cloudexpert, cloudsoftware, cloudtoepassingen, clownspieten, clunky, cmvtests, cohousing, coachingsdiploma, cocktailpragmatisme, cocoaffaire, cocokwestie, cocoleningen, cocoobligaties, cocowet, coformateur, comakijkers, comakijkt, comingofagefilm, comingofageverhaal, communicatieapps, communitystrategie, competentiescreening, complianceonderzoek, complianceteam, concentratieregime, conceptstore, concessiecultuur, conflicthuisje, consumentendatabedrijf, consumentenpiramide, consumptiestress, containerrestaurant, containerunits, containerwet, contentmarketing, contraterrorisme chef, contraterrorismediendienst, contractstage, contraterreurexcampagne, contraterrorismeexpert, contraterrorisestrategie, controlefreakerij, copypasteindustrie, copycateffect, copycatfenomeen, cornflakescafés, couplandismen, cowboyjustitie, coworkingruimte, crisisprofiel, crosschannelstrategie, crossdonatie, crossthematisch, crowdbuilding, crowdfundingconcert, crowdfundingde, crowdfundinginitiatief, crowdfunding site, crowdfunding website, crowdfundingdoel, crowdfundingexpert, crowdfunding savontuur, crowdfundingscampagne, crowdfundinginitiatief, crowdfundingplatform, crowdfundingproject, crowdfundingverhaal, crunchtime, cultcar, cultcomedy, cultuurinfuus, cultuurnetwerkers, cumulatiefplafond, cupcakewinkels, cyberafdeling, cybercomponent, cybercrimecel, cybercrime eenheid, cybermunt, cyberrisicos, cybersecurityexpert, cybersfeer, cyberwoordenboek, cyborginsectenonderzoek, dadbod, dadbodboy, dafuq, dailyracism, dakimker, daklozensjaal, dakmarktje, danceexport, dancebenadering, dansbattle, dansmulticulturalist, dashboardcamera, dataapps, dataautoriteit, datacontinuïteit, dataintensieve, databrug, datagedreven, dataminer, dataoverdraagbaarheid, dataprofilering, dataproctierecht, dataproctieverordening, dataredacteur, dateeiland, dateoefening, dateweekend, datingcamping, datinggame, datinggedrag, datingpakket, datingtour, deislamiseren, deislamisering, debatcafé, dediabolisering, deeleconomie, deeleconomiebedrijven, deelfietssysteem, deelplatformen, deelsteun, demandserie, dementieregister, denkfestival, deradicaliseerder, deradicaliserend, deradicaliseringsambtenaar, deradicaliseringsbeleid, deradicaliseringsexpert, deradicaliseringsindustrie, deradicaliseringsplannen, deradicaliseringsprogramma, deradicaliseringsteams,

deradicaliseringswerk, derdebetalerssysteem, derdebetalerkaart, derdehandscircuit, derdeleeftijdclub, designderby, designdiamantje, designerembryos, designonderzoeker, designrobot, designtriennale, desinvesteringsbeweging, detoxer, detoxobsessie, detoxweken, deuraandeuoperatie, deuraandeurtheater, devaluatiekapitalisme, diagnosechip, dialectwoordenbank, dialoogcoaches, dialoogscholen, dichtbijzorg, dienstenchequebaan, dienstenchequebedrijven, dienstenchequecircuit, dienstenchequejobs, dienstenchequesector, dienstenchequewerknemer, dienstenrobots, dierencoach, diersitter, dieselgateschandaal, diggelenkast, digilexicon, digitalemarketingspecialist, dijkmonitoringsysteem, dikdieet, dikketruienpolitiek, diplomabonificatie, diplomakloof, dispatchingdienst, dispatchingprogramma, dispatchingsruimte, divestmentbeweging, divestmentkar, divestmentverhaal, docurealityreeks, doedemocratie, doeboerderij, doelgroepenstage, doelgroepregister, donoranonimiteit, donormama, donutwentelteefje, doofpotzender, dooitraject, doorbitch, doorlijkbelasting, doorlichtingsproef, doorverkoop sites, doorverkoopwet, doorzonsteunpunt, dotcomcrash, dotcomcrisis, dotkomzeepbel, douaneapp, draag/geboortemoeder, draagmoederbeurs, draagmoederverbod, draaideurgedrag, draaitafelcultuur, dreigingsmonitor, dreigingspsychose, drieouderkind, drieouderdiscussie, drieouderschap, driestoelenprincipe, driveinfrietkot, drolemoji, droneaanval, dronebesluit, dronebranche, droneexpert, droneincidenten, droneindustrie, droneoorlog, dronetechologie, dronetelweekend, dronebestuurders, dronebeurs, dronebouwers, dronefabrikanten, dronfederatie, dronegebruik, dronegebruikers, dronelicentie, dronemilieu, droneoorlog, droneproject, dronespecialisten, dronevrij, dronewedloop, dronewereld, dronewet, dronewetgeving, druktebarometer, dubbelrichtingtrams, dubscene, duckfacepose, duimgebruik, dunnefilmproductie, durfondernemer, duurzaamheidscriteria, duurzaamheidscafé, duurzaamheidscode, duurzaamheidsdirecteur, duurzaamheidsetiket, duurzaamheidslogos, duurzaamheidswaanzin, duurzameenergiedebat, duurzameenergiereus, duurzameenergietechnologie, duurzameenergieverbruik, dvdboer, dwaalidee, dwanglening, dynamietboom, eaankopen, eboekversie, eboekenkaart, eboekenverkoop, ebookmarkt, ebookversie, ecallsysteem, echeque, ecommerceactiviteiten, ecommerceereuzen, ecommerceerein, ecommerceactiviteiten, ecommerceegigant, ecommerceegolf, ecommerceenetwerk, ecommerceereus, ecommerceesector, ecommerceesysteem, ecommerceewereld, edienstencheques, eevaluatie, efacturatie, egate, einkscherm, ekalifaat, elaughing, elearningmodule, elearningplatform, emaaltijdcheques, emaaltijdchequeskaart, emailapp, emailinterview, emanager, emodel, eonderwijs, epsychologie, esecurity, esporten, evervoer, ebolacoördinator, ebolagebieden, ebolaopleiding, ebolatestlab, ebolabestrijding, ebolacferentie, ebolacoördinator, ebolacrisis, ebolahulp, ebolapatiënt, ebolapreventie, ebolashock, ebolastam, ebolastudies, ebolatest, ebolavoorlichting, ebolavrij, ecoactivist, ecodrama, ecomagazine, ecoreactionair, ecosocialistische, ecomodernisten, ecomodernistisch, ecosportieve,

ecosysteemwaarde, egocultuur, egomassage, eisprongbewustzijn, elektronicahiphop, elektronicastijl, eliminatiedatabank, emailgate, emissiezone, emoji-film, emoji-toetsenbord, endtoendencryptie, energieblackout, energiedraaideur, energielabelloos, energievergelijker, enveloppefinanciering, epigenoomkaart, erezelfmoord, etalagescherm, etiketbrouwers, euroellende, eurouittreding, euromaaltijden, euthanasieprocedure, euthanasietraject, evenementeneconomie, evenementparkeren, eveningbeforepil, eventparking, eventagentschap, eventcultuur, eventfactor, exjihadis, excuusblog, excuusman, excuustherapeut, executiefilmpje, executievideos, experienceshoppingstraat, faalangstreductietrainer, façadedemocratie, fairplayranking, falloeraat, familiebootcamp, fancamping, farmalobby, farmaceutenlobby, farmavriendelijk, farmawhizzkid, featuringcultuur, feelgoodmomenten, feelgoodcomedy, feelgoodformat, feestdagendip, fenylaceton, festivalapp, festivaldubstep, fexamine, fietsapp, fietsbaarheid, fietskoets, fietsloopbrug, fietsmat, fietsmatproject, fietsparkeervak, fietsspiraal, filehoofdsteden, financieringshefbomen, fingerfoodproject, fingerpickinggitaar, firik, firstminutetool, fisheylens, fitnessapps, fitnessarmbanden, fitnessblogger, flamencochill, flashgeheugen, flexijob, flexibiliteitsmantra, flexibussen, flitsapp, fluoeffecten, fluodag, foefelsoftware, folterboek, foodbattle, foodcollectief, foodcultuur, foodfeestje, foodiecultuur, foodkraam, foodprofessional, foodscene, foodtruckers, foodtruckfestivals, foodtruckondernemers, formathouders, fotoapp, fotoalbums-service, fotoconcept, fotolezing, fotostok, fototsunami, fraudesoftware, freemiummodel, frequentiekam, frisdrankban, fruitmoment, fuiksites, fuikwebsite, fundraisingplatform, fundraisingproject, funkelpee, gaarkeukentraditie, gameapp, gamechangers, gameexpo, gameontwikkelaar, gameachtig, gamestester, gamestudios, gamesverdelers, gamingarcheologie, garageseks, gasbevingsland, gasroebel, gebedskaart, gebrokenflessenwijk, gehaktbalbelasting, gelegenheidsbelgotariër, gelijkkansencentrum, geloofszuil, geluidsmetingsinstallatie, geluksambassadeur, gemakssla, gematchfixed, gemismatched, genbabys, genderdiscriminatie, genderstereotypering, gendertwist, genderaanduiding, genderambivalent, gendercategorieën, genderclichés, gendergeschillen, genderjustitie, genderkamer, genderloze, gendermacht, genderoptie, genderqueer, genderregistratie, gendertransformatie, geneeskunstenaar, genenchip, genentaxi, genotsknop, genreparatie, gentledad, gentlemom, genziekten, geoenergie, geoblocking, gereedschapsbibliotheek, geslachtsneutralisering, getijdlijnd, getinderd, getrouwheidsapp, getrouwheidsplatform, geurritme, gevechtdrones, gezichtmethipsterbaardje, gezondheidsapps, gezondheidscoördinator, gezondheidsgadgets, gezondheidsshift, ggoaardappelen, ggoonderzoek, ggorijst, ggoteelt, gidsrobots, gifpilclausule, gigaindustrie, glaskoker, glasvezelbootjes, glijnummer, glitchhop, gloco, gmtechnologieën, goededoelloos, goedkopegeldbeleid, googletranslateniveau, governanceafwijking, governancecommissie, governancerapport, gpsapp, gpsapparatuur, gpsenkelband, gpsfabrikanten, gpsfunctie, gpsgegevens, gpsgestuurde, gpshorloge, gpsinformatie, gpslocatie,

gpslocatiebepaling, gpsscherm, gpstracker, graccident, grenslander, grifos, grimbo, groeibonus, groeipoolgebied, groenbedoeld, groeneenergiebedrijf, groeneenergieproducent, groeneenergierevolutie, groeneenergiesector, groenestroomcentrale, groenestroomcertificatensysteem, groenestroomfactuur, groenestroomheffing, groenestroomput, groenestroomschuld, groenestroomsubsidiëring, groenlong, groepsselfies, groepspersformance, grootouderverlof, growfunding, growkits, grungehipster, gruwelreputatie, gsmreus, haatlobby, hackerscampagne, hackingpraktijken, halaleten, halalgehuichel, halalhoek, halfjaarvisum, halloumburger, hamburgerjoint, handcarwash, handhavingswetgeving, hardcorejihadi, hardloopapp, hardloopeconomie, hardrockdubstep, hardwarekennis, heimweeonderzoeker, helpiecampagne, helpiereclamespotje, helpiespot, herdenkingsprofiel, herframen, herkomstcheck, hernieuwbouw, heropvoedingscampagne, hfthandel, hftpartij, highschoolsitcom, hightechjob, hightechlandschap, hightechwerknemer, hijabverbod, hippegranenschap, hipsteraanwezigheid, hipsterbaard, hipsterbankier, hipsterbril, hipstercafé, hipstercultuur, hipsterdecadentie, hipsterding, hipsterenclaves, hipsterhaat, hipsterkoning, hipsterkot, hipsterorgasme, hipsteroverdosis, hipsterpak, hipsterplatform, hipsterpolitie, hipsterpubliek, hipstershit, hipsteruniform, hipstervriendinnetje, hipsterwereld, hitparadehiphop, hittestressindex, hivpolikliniek, hogesnelheidshandel, hogesnelheidshandelaren, holebiacceptatie, holebiactivisten, holebiadoptie, holebimonument, holebiorganisaties, holebiouderschap, holebiserie, holebistellen, holebiuitgeverij, holebivriendelijke, holebiboek, holebiboekhandel, holebiliteratuur, holebiseks, holebithemas, homegrown, homoapp, homoemancipator, homoapp, homogetrouwd, hondencatering, hoponhopoffbiljet, horecaboy, hormonenbiefstuk, horrorjihadisme, horrorovertredingen, hottub, hotdamn, hotdogkorstpizza, hotelalstuin, hotelboekingsapp, hotelzoekmachine, hotseat, hotseknotslanding, housemuziekfestival, houtjetouwtjebegroting, hufi, huidtuberculinetest, huistuinkeukenislam, huisartsenwachtregeling, hulptv, humaninterestprogramma, huppelpop, huurindexdiscussie, huuroffer, hygiënehypothese, hyperlift, ivoting, icebucketchallenge, ideeëneconomie, ideeënopera, ideologiecampagne, iedervoorzichmentaliteit, immersieonderwijs, importyuppen, ingameaankopen, inoutrereferendum, inbreidingsproject, incubatortraject, indexknuffelaar, indieboyband, indielabelgroep, indierockband, indiegod, indiepopcombo, indieproducer, indierockparadijs, indiespellen, indoctrinatieprogramma, inentingsvoorlichting, inhoudsmens, injectieteller, injectiewater, inkomensbegrenzing, inleefhuis, inschakelingsuitkering, insectenbedrijf, insectenhotelletjes, insectenmeel, insectenproductie, inspiratieshot, instagramalbum, instagramfeed, instagramfotos, installatieperformance, instaproef, integratiecirkel, integratietoelage, integratieverklaring, internetdominee, internetgamingstoornis, internetkolossen, internetpreken, irritatieindex, islambeurs, islamdesk, islamexpertise, islamoverheersing, islamspecial, islamstickeraffaire, itbeveiligingsbedrijf, ittechnicus, jaarwoordenzoeker, jadicalisejing, januaristress, jarenvijftigvljijt, jeugdhandhaver,

jezidihoofdstad, jezidiman, jezidiminderheid, jezidivrouwen, jihadexpert, jihadonderzoeker, jihadtoeristen, jihadbruiden, jihadbruidjes, jihadcel, jihadicentrum, jihadiexpert, jihadifabriek, jihadigroepen, jihadimoskee, jihaditrip, jihadiversie, jihadiwereld, jihadisalafisme, jihadistenbeweging, jihadistendoctrine, jihadistenleger, jihadistennaam, jihadistennetwerken, jihadkanker, jihadkenners, jihadkoppel, jihadoloog, jihadpropaganda, jihadranselaar, jihaduitspraak, jihadverdachte, jihadwatcher, jihadzaken, jobcafé, jobmarketeer, joemelsoftware, jongehondenzender, jpgfotos, juichpak, junglefunk, kaaimantaks, kadoosje, kajaktivisten, kameleontaks, kamikazecoalitie, kamikazeinterview, kampeerautoplaats, kansarmoedeindicatoren, kansenwijk, kantelcliquet, karaataks, karaktermonster, katholiekligh, keerklepregeling, keldermoskeeën, keuzestress, keuzejeuk, keynotespeeches, kibbelparlement, kiespasje, kijkcijferbom, kijkcijferstress, killerapp, kilometerplafond, kilowattuurteller, kimojis, kindertijdeten, kirigami, kirigamikunst, kirigamikunstenaar, kissandridezone, klikcultuur, klimaatbakfiets, klimaatdorp, klimaatefficiënt, klimaatmars, klimaatobligatie, klimaatprestatie, klimaatrijst, klok bom, klokkenluidersaccount, klokkenluidersdesk, klokkenluiderwet, knelpuntberoep, knelpuntberoepenprobleem, knipperlichtoefening, knoopsgatchirurgie, knopologie, knopoloog, knuffelconcert, knuffellunch, koekjesmuur, koestertas, koffiecupoorlog, koosjerheidscertificaat, kopieermaatschappij, korteberichtenplatform, korteberichtjesdienst, krautrockbeat, kredietshoppen, krekelkroket, kruisvaarderscoalitie, krumpin, kunstenaarsbiopic, kunsthotspot, kwantumcomputeronderzoek, labaneh, labelinflatie, labelitis, ladyfriendly, lageemissiezone, lagekostengevecht, lagekostenorganisatie, lageprijswinkel, lagerentepolitiek, lagerenteklimaat, lanceringsevenement, landdieet, laptopdjs, laptopbeat, laserlichtbron, lasermicroscoop, lasershoot, lastminutecadeautje, lastminuteopdracht, latenighttalkshow, latinocinema, leadout, ledlampen, leegstandsbelasting, legestoelpolitiek, leisurekledij, lethalwapens, leunstoelbetrokkenheid, levenscyclusdenken, levenseindepil, levensstijlziektes, levenstestamentregister, libidopil, liborfraude, lichteditie, liefdesslotje, liefdesslotjeshek, lifeloggen, lifestyleevenement, lifestyleindustrie, lifestyleproduct, lifestylekeuken, lighttaks, likeknop, lipsyncappje, livestreaming, livestreamingmarkt, livestreampje, locatieapp, lokautoarrest, longstayafdeling, longstayinstelling, loombandjes, loonnaarprestatieprincipe, loselosesituatie, losloopbos, losloopplek, loungerestaurantketen, lowtopsneakers, lowbudgetgebouw, lowbudgetmuseum, lowcostbeursmakelaar, lowcostmentaliteit, lowcostoperatoren, lowcostproject, luchtterrassen, luisterworkshops, lumbersexual, luxeziekte, lycrabroek, maaltijdaanbieder, maatwerkdiploma, magneetlift, magneetscanner, mailhygiëne, mainstreammedia, mainstreamkarakter, mainstreamkop, mainstreamradios, makeoverprogramma, makkar, mantelzorgforfait, marshmallowtheorie, matchfixingschandaal, mayonaisewet, mediablackout, meditatieapp, meeneemsalades, meeroudergezag, meerouderschap, meeuwenproof, megagaleriecomplex, megabeeldscherm, megaset, megavorm, meldingsknop,

menstratiecup, menstruatiecups, merkelt, mersvirus, messageapp, messensterilisator, metahumor, metroapp, middelvingeraffaire, middeninkomensval, middenvingerpartij, mierikswortelmelk, migrantenboten, migrantenhinder, migratiecrisis, migratieeisen, migratieuitdaging, migratieconflicten, migratiemoe, migratieroots, migratiestorm, migratiewoordvoerder, milieuscreening, millennialgeneratie, mindfulnessmeditatie, mindfulnesstrainingen, miniclips, minilabs, minibustaxis, mirabegron, mixdrankjes, mobielebetaalapp, mobieletelefoniedata, mobielelefoontijdperk, mobilhomeparking, mobiliteitoplossend, mobiliteitsapp, mobiliteitsprocedure, mobiliteitsrevolutie, mobiliteitscore, mocktailwagen, mockumentaire, mocrooorlog, moestuingoeroe, moestuinmythe, moestuinopstelling, moestuinromantiek, mondgasme, monstertruckongeluk, moordhuwelijkarrest, morfinebrood, moslimterreurgroepering, mozaïekfilm, mp3geluid, muisklikcultuur, multicultiplatform, multikopter, multimediatriller, multiscreenomgeving, multisportzone, muntjesproef, muziekapp, muziekherkenningsapps, muziekstreamer, muziekstreamingmarkt, naaktselfie, nabijheidssensor, nachtformaat, nachtnanny, nanodebat, nanostofje, napolatte, narcoflik, navigatieapp, nearmiss, neereferendum, netheidspersoneel, neutronenscan, neveldouche, newspeakbetekenis, newwaveromantiek, nicheacts, nichegeluid, nietbio, nietomroepen, nieuwsapp, nieuwstress, nimbycliché, ninetofivetype, noflylist, nononsensecultuur, nononsensemanager, nononsensestype, noisepopband, nomadengeneratie, nonfooddiscounter, nonplace, nonplek, noodwijk, noveltyhit, nufolk, nudgebeweging, nudgeteam, nuluitstootauto, nummerplaatcamera, obesitashoogleraar, oefendrone, oeryoga, oertracker, oeryup, oesterpicknick, offlinegebruik, oldskoolhiphop, olieshampoo, olifantenrusthuis, omacool, omaregel, omgevingssensitiviteit, omnummerdienst, omnummertool, ondemandpakket, ondemandradio, onduurzaam, oncofertiliteit, ondernemerspagina, onderwaterdrones, onderzoekersjournalist, onemanband, onkostenapp, onkruidtrein, onlife, onlifewereld, onlineabonnee, onlineactie, onlineactiviteiten, onlineadvertenties, onlineadvertentiewerving, onlineapotheken, onlineartikels, onlineautoverkoop, onlinebedrijfsspionage, onlinebetaalplatform, onlinebetaalservice, onlinebetaalzender, onlinebibliotheken, onlinecontent, onlinefora, onlineformulier, onlinegroepen, onlineidentificatiesysteem, onlineindustrie, onlineinschrijfsysteem, onlinekanalen, onlineleefwereld, onlinemoskeeën, onlinenetwerk, onlineondernemers, onlineplatform, onlinesupermarkt, onlinetaalprogramma, onlinetool, onlinetv, onlinevrienden, onlinewinkel, onlinewinkelen, onlinebeurshandelaar, onlinebezoek, onlineboodschappenservice, onlinebusinessmodel, onlinecarrièreplanning, onlinecriminaliteit, onlinedebatten, onlinedeelplatform, onlinedistributiemodel, onlinefilmpremière, onlineformule, onlinekruidenier, onlineleveranciers, onlinemanifest, onlinepersconferentie, onlineprogrammas, onlinerelease, onlinesessie, onlinetaxidienst, onlinetrading, onlinetransactie, onlinevergaderingen, onlineverkoopmarkt, onlinewebwinkel, onlinewinkelservice, onlinewoordenlijst, onlyabonnement, onneukbaar, ontbijtgranenbar, ontdieseling, ontdieselt,

onthoofdingsflimpje, onthoofdingsvideos, ontplofteparachuteoutfit, ontwikkelingsperspectiefbaan, opblaasverblijf, openaccess, openaccesstijdschriften, opengrenzenbeleid, openarmenbeleid, openluchtboekhouder, opensourceplatforms, opensourceprogramma, openwereldspel, opinieprogrammas, opiumbier, opiumgist, opleidingsrekening, oppaspoule, opportuniteitsonderzoek, oppositieeffect, optinprincipe, optoutmogelijkheid, ouderdomspakken, ouderdomssimulatiepak, outofficecommunicatie, ovchippoortjes, overaged, overgeefmachine, overspelwebsite, paardenvisserskoppel, paddenstoelenbroed, padelclub, padelcourt, pakjesboxen, paleoom, paniekcoach, parcelshop, participatiecampagne, participatiekaart, participatiemakelaar, partygebruik, partyrijtuig, pastafarianen, patiëntendata, patsertaks, pauzereclame, payrollcontracten, pcsyndroom, peertopeerssoftware, pegidler, pensioeneiland, peplumjurk, perceptieoorlog, petitesite, petroleumdollar, pgbaanvragen, pgb rapport, phishingbende, phishingdossier, phishingwebsite, photoshopbehandeling, piëteitszone, pinupmodel, pitstopcultuur, pixelposter, pixelstad, plafondspaarformule, plafondsparen, plantenasiel, plasapp, platformneutraliteit, plattevijgensocialisme, playsuit, plofslak, pluginhybrid, podcastcultuur, podcastiaans, podcastserie, podcastwoestijn, poepemoji, poepdonatie, poldercomplex, polderjihadisme, polderomerta, politiedrone, polkadotbikini, poortjesspringers, popartlegende, popjihadi, popupbezorgrestaurant, popupinitiatief, popupminigolfterrein, popuprestaurant, popupbar, popupboetiek, popupcafé, popupcinema, popupconcept, popupcorrespondenten, popupeiland, popupevenementen, popupgrillrestaurant, popuphotel, popupmarkt, popupschool, popupshows, popupsporthal, popupstreet, popupvestiging, popartkleding, popcarnaval, porffoliodirector, pornocasting, positieupdate, postfeestdagendip, postitbriefje, posterboyimago, postgender, postpaidaanbod, postpaidsegment, postpunkgenre, potpourriwetten, powermetalband, powerpointpresentaties, preawareness, preroll, presmartphone, prestreamingtijdperk, precommissie, prepack, prepaidsimkaarten, prestatiewereldje, prewaarborgregeling, primetimeprogrammering, printmagazine, printpublicatie, privateequityeigenaar, privateequitybazen, privateequitybedrijf, privateequityfinancier, privateequityfirma, privateequityhuis, privateequitymaatschappij, privateequityspeler, privateequitystijl, privateequitywereld, privébeveiligingsagenten, promoelpee, promobezoekje, promofotos, promotrailer, promposal, prosumententarief, proxyoorlogen, proxywar, prullenbakscholen, psychfolk, psychotainment, puberdecoratie, pushback, pushberichten, pushbackbeleid, pushnotificatie, qatverslaving, queererotiek, quinoasalade, quinoakut, racismerel, radicaliseringsdienst, radicaliseringsprobleem, radicaliseringsproblematiek, radicaliseringsprogrammas, radiopodcast, ramenbar, reboot, retweeter, reaguurbot, reaguursel, realityspelprogramma, realitydocus, reallifedocumentaire, realtimeverkeersinformatie, reclametweets, recyclagecijfer, recyclepullenspeeltuin, refofilm, regenboogmonument, regenboogpiet, regenboogsteegje, reisbeoordelingsite, rekencoaches, relingos, rendementsdenker, repeatmodus, rescops, responsteam, restaurantbeoordelingsite,

restaurantreserveringsite, retroartiest, retrostemming, reviewcommissie, ridesharing, rijgedragapp, rijgedragsoftware, ritjesapp, roamingtarieven, roamingtoeslag, robotadviesraad, robotjournalist, robotmedia, robotpatrouillewagens, robotvlinder, robotwapenwedloop, rockerij, rockfeminisme, rodebietenhummus, roetpieten, roetveegvarianten, romcomgeluk, rondtweette, rotatievoetbal, rôtitsoire, rugbyrolstoel, ruiltrend, runshoppingzone, salafijihadisme, samaritaanmasturbatie, samentuinproject, samenzweringswebsite, sandwichbeurs, sandwichtoemaat, sarsuitbraak, savantautist, scanautos, schalieolie, schaliegasproject, schalieproductie, schaliesector, schandaalbestseller, scharrelslakken, schedeldaktransplantatie, scheidingsambtenaar, scheurtjesreactor, schijnondernemerschap, schoolkeuzestress, scifiklassieker, screeningsautoriteit, scriptedrealityreeks, securityblog, seksapp, seksismewaakhond, selectietool, selfassessment, selfbankautomaat, selfiecultuur, selfieepidemie, selfietijd, selfietijdperk, selfietrui, selfiewedstrijd, selfiecultuur, selfiehoofdstad, selfiejagers, selfiekoningin, selfiemuseum, selfieprotest, selfiestunts, selfietime, selfieverbod, selfiewoede, seniorenmomentje, septemberdip, sfeertechno, shakenbabysyndroom, shariawet, shishalounges, shufflemodus, singletasken, sitcomervaring, sjoemelbedrijfje, sjoemelcode, sjoemelfraude, sjoemelschandaal, sjoemelsoftwareschandalen, sjoemelvis, sjoemelwagen, skypeinterview, slaapbankgate, slaapconsultant, slackerreputatie, slapscreen, slavenhandelmogendheid, sleeppod, sleepnetactieken, slowburneffect, slowbeat, smdatingsite, smartlight, smartphoneafdeling, smartphoneapp, smartphoneapplicatie, smartphoneeigenaar, smartphonefabrikanten, smartphonefotos, smartphonegames, smartphonegamma, smartphoneindustrie, smartphonemaker, smartphonemarkt, smartphoneplatform, smartphonerevolutie, smartphonetijdperk, smartphoneverkoop, smartphonebetaaldienst, smartphonecamera, smartphoneformaat, smartphonefotos, smartphonegegevens, smartphonegigant, smartphonegolf, smartphoneversie, smartwatchhype, smartwatchindustrie, smartwatchmarkt, smartwatchkast, smartwatchplan, smeergeldtraject, smileyzonsverduistering, smileyachtig, smsapp, msgate, smsstrook, smswandelstrook, snacktelevisie, snapchatters, snelkeuzetelefoonnummer, snelleresponsteams, soapfactor, soapieschrijver, socialemediakanaal, socialemediamoord, socialemediasite, socialemediabedrijf, socialemediagebruikers, socialemedianetwerk, soulsirene, spaarplafond, spankingforum, speelpleincoördinator, speelpleinorganisator, spelletjesapps, speltstroop, spinofffonds, spinningavond, spoedeuthanasie, spookcorner, spookjongeren, spooktrap, sportapp, sportleggings, sportsensor, spraakapp, sprinkhanenfondsen, spyspoof, staartstrop, staatshackers, stadsapps, stadsnatuurkaart, stadsnatuurwerk, stadsvarkens, standupgoochelaar, standaardemail, standaardapps, startupcultuur, startupwedstrijd, startupbedrijven, startupmentaliteit, stedentheater, stemboot, sterfbureau, steunplafond, stiltehuisje, stiltemassage, stilteplekje, stilteweekend, stokachterdedeuraanpak, stopformulieren, stopmotionanimatie, stopzettingsverzekering, straatatelier, straatkunstproject, straatmeubilaircontract,

straatpoëten, straatproza, streamingaanbieders, streamingbusiness, streaminghitlijst, streamingmuziek, streamingoorlog, streamingsites, streamingbedrijf, streamingcijfers, streamingformule, streaminggewijs, streamingmogelijkheid, streamingpionier, streamingreus, streamingsapp, streamingspecialist, streamingstrijd, streamingsucces, streamingverhaal, streamingverkeer, streetart, streetartkaart, streetartkunstenaars, streetfoodzaken, stressspecialiste, stresstest, stressdokter, stripbios, stripbiografieën, strippark, stroopwafelpieten, studeerbubbel, subprimehypotheekverstrekker, subprimehypotheek, subprimerisico, subreddits, subsidiejager, suikerkanon, supercoachapp, superfancy, surroundervaring, swapcontracten, symfrock, taal oefenplek, tableteditie, tabletmodus, tabletgeneratie, takscificatie, tampons, tapiocawrap, taxonweb, taxonwebaanpak, taxiapp, taxiappbedrijf, taxiappoorlog, taxiapplicatie, taxiappmarkt, taxistartup, taxshiftakkoord, teambuildingsmoment, techzeepbel, techneck, technologieblogs, techwerk, tefalpremier, teflonpoliticus, telefoonapp, telefoonflits, telefoonhokjeshumor, telefoonvrij, telepresencerobot, tentgewijs, tentoonstellingsbox, terraskruiden, terreurstrijder, terroroehoe, terrorclown, terugbetalingsplafond, terughackwet, terugkeerloket, terugnameakkoorden, theatercollege, thuisbatterij, thuishandboek, tiltul, timein, timeoutprojecten, timelapsevideo, toegangsbevoegdheden, toegangsknop, toekomstgroente, toerismemagneet, toolbelt, toptrack, touchscreenschermpje, trackandtracesysteem, trackingsoftware, trackingtechnologie, trancefolk, trancedeque, tranentaart, transactrice, transman, transfertarget, transgenderverpleegkundige, transgendermodel, transgenderparade, transgenderwetgeving, transmensen, transparantiewaakhond, traphiphop, treinapps, treurpark, troeteljihadi, troetelrepubliek, trojka, trojkabeleid, trouwproeverij, trumpisme, tuningmechanisme, turnarounds, tvspinoff, tvstreamingformule, tweepersonenindecockpitmaatregel, tweepersonenindecockpitregel, tweedepijlerpensioen, tweetfilosoof, tweetmeme, tweetplicht, twitteraccount, twitterfeed, twitterfotos, twitterkoning, twittersfeer, twitterrecord, ubbleleeftijd, überfrau, überhippe, uberification, uberisering, übermacho, übernerdy, übernacht, überromantische, überslacker, uitgaansbeleidscoördinator, uitgiftegesprek, uithuilplek, uitschakelingsuitkering, uitzetlocatie, ultrahd, undercoverimam, upcyclingbeweging, vacuümtoiletten, vaderschapsrust, vannegentotvijfbaan, vastebanenwet, vastgoedbubble, vastgoedcoach, vechtagent, veelplaatser, vegabiefstuk, vegenpiet, veggieradicalisten, veggiebox, veloverbeenrage, veranderingsflow, verappartementisering, verbetertermijn, verhalenworkshop, verhipstering, verhuurwebsite, verkeersapplicatie, verkeersdatabank, verkrottingsboete, verliesverliessituatie, vernatuur, verpakkingsvrij, versleutelingsprogramma's, versunit, vertalersslam, vetpact, viagercontract, viagersysteem, viagerverkoop, videoarbiter, videoondemanddienst, videoconferentietoepassing, videogamedesigner, videogamesite, videosolliciteren, videostreamingapp, videostreamingsdienst, vierseizoenenborder, vignettesfilm, vintagetrend, virtualrealitybril, virtuelerealeitsbril, vleerdino, vlinderboerderijen,

vluchtelingendiscours, vluchtelingenhaat, vluchtelingen spreiding, vluchtelingen vrij, vluchtenlingendrama, vluchtlingencrisis, voetbalvluchteling, vogelvaartocht, volgcookies, voogdagent, vooruitgangsliberalisme, vormgevingssite, vrproductie, vrspelletjes, vredespianist, vreemdgangerssite, vreemdgangerswebsite, vriendschapssites, vrouwenwebsite, vuilbakkentaks, vuilnisbakkentaks, waakzaamheidscampagne, waakzaamheidsnota, wachtrijentertainment, wachttijdindicatie, wagyuburger, wangslimvliestest, warmvatten, waszolder, waterblackout, waterpasapp, wauwcultuur, webapp, webdrugsmarkt, webhostingbureau, webradiokanalen, webtooltje, wegdrammen, weggeeftafel, weggeswipet, wegwerpdrone, wegwerpstrategie, weldoenerscomplex, wellnessdeel, welstandsvlucht, welstandsvlucht, welvaartzieken, welzijnsgevoeligheid, wereldjihad, werkmailtjes, whatsappberichten, whatsappcontact, whoiswho, wiewerblog, wietbank, wietkluis, wietnota, wietstad, wietteeltspecialist, wifiaanbod, wifiapparatuur, wifiband, wifihotspot, wifiinstallaties, wifilogo, wifimodem, wifinetwerk, wifiprotocol, wifipunt, wifisignaal, wifisnelheid, wifitoets, wifiverbinding, wifiverbod, wifizendertje, wifizone, wijenzijsamenleving, wijenzijverhaal, wijtegenzijdiscussie, wij/zijperceptie, wijkservicemanager, wilddetectiesysteem, wilddetectiesysteem, wildplukwandeling, winwinstfeertje, winwinprojecten, windclaim, windhanenbeleid, winkelsfeergebied, wittejongensgroove, woonspaarrekening, woordenfluisteraar, wormenpillen, wouldbejihadist, wowmoment, wraakpornozaak, yacon, yezidimeisjes, yeziditiener, yezidische, yogahype, yuppensport, zaadbommetjes, zangerrapper, zeewierburger, zelfdodingsknop, zelfmaakwebshop, zelfmoordknop, zeroafvalcultuur, zerohero startup, zerotoleranceaanpak, zijinstromers, zitziekte, zoetwaterzwemkaart, zoldermuseum, zonneenergieparken, zonneenergiesteun, zonneenergiesubsidiëring, zonnestroomorganisatie, zonnestroomorganisatie, zorgrobot, zorgmanege, zuurdesembakker, zuurdesempizza, zwakbodemmuziek, zwartwitdebat, zwartwittepieten, zwartepietdebat, zwartepietdiscussie, zweefdino, zwembadsmaak, zwerfbibliotheek, zwerfvuilberg, zwermttechnologie, zwijgdeals, zzp-bestaan, zzp-constructies, zzp-elite, zzp-kennisplatform, zzp-organisaties, zzp-probleem, zzp-schap

Niet behouden neologismen uit Neoloog (N=828)

0800-nummer, 1%-regeling, 100-procentscontrole, 15-aprilarresten, 2016-modellen, 24-ureneconomie, 24-uursbegeleiding, 24-uurskanaal, 24-uursopvang, 24-uursstaking, 24-uurszender, 30-urenweek, 360-gradenevaluaties, 365-dagen-per-jaar-economie, 3x3-mannen, 3x3-vrouwen, aaa-haaa!-koortjes, aanbodsmerken, actuagasten, adoptie-ambtenaar, advocatenblogs, afgesponsord, age-roman, age'-documentaire, agree'-knop, agro-entrepreneurs, agro-marketing, alarmbelgesprekken, algebra-analfabeten, alibi-denken, alienjagers, all-you-can-eat-sushirestaurants, anorexiamannequins, anti-bedelwetten, anti-corruptie-operatie, anti-establishmentpartij, anti-eurogroep, anti-eurostemmers, anti-homowetten, anti-immigrantenpartij, anti-immigrantenretoriek, anti-islamisten, anti-jihadprojecten, anti-

moslim, anti-moslimadvertenties, anti-moslimdemonstraties, anti-moslimhordes, anti-
psychiatrie, anti-terreuractie, anti-terreurcel, anti-terreurdiensten, anti-
terreuroperaties, anti-terreurorganisatie, anti-terreurrechter, anti-wit,
antibioticaresistent, antidaklozenspikes, antidroneboerka, antimigratieklimaat,
antiterreur-eenheid, antiterreur-wet, antiterrorisme-expert, antitsunamimuren,
antiverdwaalbandjes, aperitiefbites, app-industrie, app-muziekjes, app-ontwikkeling,
app-vriendelijker, app-winkeliërs, appbare, arbeidsmarktoriëntatie,
arbeidsverminderingssystemen, armoedeschachten, arty-farty, asbakzakken,
asfaltgieters, asielgemeentes, asielindexen, asielquota, aspirant-asielzoekers,
aspirant-jihadisten, atolnaties, autismespectrumdiagnoses, avocado-emoji, azc-
bestormers, azc-haters, b&b, baanwinkelgebieden, baanwinkelprojecten,
baardnetjes, babyfora, bail-outmaatregelen, balgouchelt, balkstenen, bankvirussen,
baviaanbroekjes, bbp-deflator, bed-bad-brood, bed-bad-brood'-principe, bed-bad-
en, belcafés, beleidscabaret, belevingsateliers, belevingsgehalte, belgotarische,
beloningsstormpje, berechtingskoffertje, betaaldiëten, betredingsprocedures,
beurspensionado's, bewakingsdrones, bewustmakingsorganisaties, bff's, bier-
architecten, bierfietsregels, binge-kijken, binge-perspectief, bingeforums, bio-
ethische, biotechbedrijfjes, bitcoin-gebruikers, bitcoin-puristen,
bitcoinoverschrijvingen, blauwhelm-baby's, bleitrol, blijdschapspillen, bling-babes,
blockblusters, blockbuster-franchises, blockbuster-technologieën, blockchain-
cursussen, blockchain-investeerders, bloedjuwelen, bloemkoolchips, bluetooth-
systemen, bluetooth-wagenkits, boekencafé, boerenkooljus, bonenballetjes, bonus-
cd, bonusinners, botoxproducten, bottom-up-projecten, break-even'-regels,
breedband-internet, brontoburgers, brood'-discussie, brood'-kwestie, brood'-opvang,
bullit'-treinen, bunga'-feestjes, bureaustoelracete, business-to-business-
communicatie, buurderijverantwoordelijke, caloriecorrecte, cam-versie,
campagnefighters, cannabisbehandelingen, capaciteitsgemeenten, cdo's,
champagnevaartjes, charme-offensieven, chattaaldeskundige, chocoladegranen,
citroenjobs, class'-aanpak, cloud-gebaseerde, cloud'-diensten, cmv-infectie, co-
creëren, coco-gate, coconjournalisten, collega-separatisten, comiclezers,
communiebarbecues, conflictregio's, consumentenzoektermen,
contraterrorismewetten, copycat-producten, cosmetica-wetgeving, cover-your-ass-
rules, crisiswinkeltje, cross-dressers, cross-overcinema's, crowdsurftaferelen,
crowdfundingprojecten, cut-up, cutting-edge, cyberclichés, cybergebeuren,
cyberrelletjes, cybersecurityregels, cyborg'-insecten, cyborgkakkerlakken, d-woord,
dadbod-jongens, dagboekpoëzie, dance-artiesten, dance-festival, dance-groep,
dance-industrie, dataconnectie, dataminingprojecten, datazoekende, date-fondsens,
datr'-cookie, db+, deeltjessproeiers, deëscalatiewetten, deradicalisering,
deradicaliseringmaatregelen, deradicaliseringsacties, deradicaliseringsbureautjes,
deradicaliseringsdeskundige, dertigersvraagstukken, designerembryo's,
designtijdschrijft, dieetfabrikanten, dieetverkopers, dieselgate'-wagen,
digitaalvriendelijke, digitaletelevisiepakketten, dijkankers, dinovrouw, disco-
jongleren, dislike'-knop, doelgroep-gezinnen, doellijnbeslissingen, donorbarbecues,
donorprofielen, doorburgeraars, doorklik-aankopen, doorlooppassages, doorsnee-
zpp'ers, draagmoederkandidaten, drachmisten, drone-geluiden, drone-opnamen,
droneconcept, droneverhuurders, drugscoördinator, dubstep-plaat, duckfacefoto's,
duurzaamheidstudies, duurzaamheidswetenschappen, duurzame-
energiecoöperaties, e-auto's, e-betalingen, e-bike, blijkt, e-bookclausules, e-
commercebedrijf, e-commercejobs, e-commerceopslagplaatsen, e-commercewerk,

e-consumenten, e-contracten, e-klanten, e-mailmeldingen, e-ondernemer, e-versie, e'-logo, ebola-donors, ebola-epidemie, ebola-uitbraak, ebola'-boodschap, ebola'-campagne, ebolacomplicaties, eco-vriendelijk, eco-warriors, eco-werkers, ecostadsprojecten, ééneuromaaltijden, ego-documentaire, eindejaarsstickers, electro'-stijl, elektriciteitsprijskloof, elektrojasjes, emotiepoppetjes, end-to-end, end-to-end-oplossingen, euro-maaltijden, euthanasie-aanvraag, euthanasie-expert, euthanasie-uitvoering, euthanasiedebacle, evidence-based, excuusbelgen, extreem-rechts, fair-play, fairplayscores, fakebeelden, falafelballen, fantasy-prone-persoonlijkheden, fantasyvideogames, farmagroeier, featuring'-golf, fertiliteitsziekenhuizen, festivalambtenaren, festivalmunten, fietsknooppunters, fietstaxistandplaatsen, fietsvaardigheidstrainingen, filesluipers, flash-geheugenbanken, flirtcompetenties, food-evenement, freecycle-groepen, freemium'-model, gadgetstores, game-consoles, gamecompetities, games-franchises, gaming-industrie, gecoproduced, geestesziekte-genen, geleukt, geloodgieterd, geluidsfietzen, geluksprofeten, gemeenschaps-wachten, geminioniseerde, gendergelijkheid, gendergrapjes, genderklachten, gendermoorden, gentech-mais, geo-engineering, gephotobombed, geregramd, gesextingd, geslachtswisselaars, gettoleningen, gewenningsinstallaties, ggo-bintje, ggo-maïs, ggo-populieren, glasvezeloptica, glazenwasserrobots, gm-rijst, gm-voedselgewassen, goth-ouders, governance-punten, governance-studie, gps-operatoren, gps-posities, gps-satellieten, gps-zender, gps'en, groene-stroomcoöperatieven, groenestroom-certificaten, gruwelijhadisten, gruwelnormen, gsm-antennes, gsm-infrastructuur, halal-eitjes, hechtingsuurtje, hernieuwbare-energiecoöperatieven, heroïneborden, herstructureringscao, hft-handelaren, hft'ers, hipsterambities, hipsterboys, hipstermillennials, hipstervrienden, hipstervriendinnen, hit-and-run, hit-and-runaanvallen, hitlijstkopieën, hitlijstrappers, hiv-migranten, hobby-detective, hoge-frequentiecamera's, hokjesgerichte, holebi-gemeenschap, holebi-rechten, holebi-zijn, homo-ouders, homo-pesten, homofobie-vrije, hondenresorts, hoogrisico-organisaties, horrorhuishoudens, hot-spot, housesit, houtinvloeden, hub-and-spoke, hulptools, huurbaarmoeders, hygiëne-hypothese, ideeënpark, ijssnelwegen, ijssnelwegroutes, ikea'en, image-building, immigrantenclichés, indie-tijdgenoten, indieblues, indiekrakers, informatie-overload, infotainmentbladen, insectenproducenten, instantcamera's, integratiefrustraties, internet-ondernemers, internet-televisie, internet-winkelen, internetbril, internetfenomen, internetforaschrijvers, internetpenetratiegraad, islam-leerkrachten, islamtak, it-girl, ivf-kliniek, ivf-methode, ivf-techniek, jezidi-buren, jihad-aspiranten, jihad-leuzen, jihadfinanciers, jihadi-infanteristen, jihadi-processen, jihadisten-vleugels, jihadistenregime, jihadstreken, jihadzorgen, joint-venture, kalender-app, kamikaze-regering, kentekencamera's, kernwapenpacifisten, kinderkasseien, kirigami-artiesten, kirigami-cellen, kiss&ride-zone, kleurpieten, klikkers'-telefoonlijn, klimaatfietsers, klokkenluidersmailtje, kloon-apps, klusappartementen, knipperlicht-vriendin, knipperlichtprovisies, knuffelvriendelijke, kook-tv, kunstvagina's, kweekvlinders, lage-emissie-zones, lage-inkomensgezinnen, lagekostengiganten, lakrobots, lasercamera's, laserraptors, led-schermen, led-technologie, led-verlichting, ledtoepassingen, lees-app, leuk'-duimpje, levenskaarsen, life'-verhalen, lifecycle-fondsen, lifecycle-producten, lifestylevlogs, light-dranken, lijstkapingen, like'-knop, live-action, livevideostreams, livevisuals, lokmaatregelen, lol-gebruikers, lol'ers, lonewolfaanvallen, loopbaan-begeleiding, luchtviolisten, luchtwaardigheidsnormen, m'as-tu-vusfeertje, mainstreamalbums, mainstreamimago, malware-codes, man-

manrelaties, mannenrokjes, marathon-uitzending, massa-individualisering, masterleerkrachten, meerderwaardigheidscomplex, meerkatters, meerwaarde-belasting, mega-installaties, mega-investeringen, megacamping, megawindstoten, meningentsunami, meta-films, metacinema, microloggers, midcals, middelvinger-emoji, middenveld-organisaties, middenveldinitiatieven, migratietankers, migratiexperts, millennial-ouders, min-30'er, mini-crash, moddermails, modemtechnologie, moeder-en-baby-hotels, moestuinbenodigdheden, moestuinklanten, moestuinverwachtingen, mongo's, moslimsnacks, multimedia-dagboek, multisportevent, nagelstickers, nanosensoren, navelshirtje, netheidsambassadeurs, netwerkcoördinatoren, new-agemuziek, newwavesynthesizers, niche-producten, nichefondsen, nicheliteratuur, niet-euroland, no-fly-zone, no-nonsense-beleid, no-nonsenseteam, noisefreaks, noisepunkbands, noiserock-riffs, non-foodketen, non-probleem, non-spot'-reclame, non-strategie, nucleo's, nul-belasting, nul-urencontractanten, nultwintiggers, old-school, oledpixels, omnisportfederatie, onco-fertiliteitsarts, oneshotmovies, online-betalingsmechanismen, online-boeven, online-handel, online-jihadi's, online-producten, online-publicatie, online-trendspotters, online-wereld, online-wervingscampagnes, online-winkelformules, onlinebesteldiensten, onlinefenomeen, onlineliefde, onlinemachten, onlinenavigatiekaarten, onlinerechten, onlinetijden, ontgenderen, open-armen-politiek, openluchtcinema's, optimalisatietools, ouderprofielen, outplacement-begeleiding, outsourcingoperaties, paraplustations, partytrucks, passagierstolpoortjes, pauze-pancartes, pc-tijdperk, peer-to-peer-financieringsplatformen, peer-to-peerinitiatieven, pgb-problemen, pilot-aflevering, pitchfilmpje, poëziedokter, pop-up'-element, pop-upontmoetingssessies, pornadeskundige, post-doc, post-paid, post-punk, postpaid-klanten, pps-constructies, premiumsmartphones, prepaid-kaart, pretrappers, private-equityfirma's, private-equitytak, privéagenten, privéappjes, privémobieltjes, pro-actieve, pro-euro, productiebelastingen, programmeerinstellingen, promo-activiteiten, promo-clip, proxy's, puntjesboeken, push-notificaties, pushbackboten, r&b, r&b-collega's, r&b-generatie, r&b-liedje, r&b-prinses, r&b-queen, r&b-ritme, r&b-ster, r&b-zangeressen, r&d, r&d-budget, radio-airplay, radio-en-podcastapp, rags-to-riches'-verhaal, railbikers, rawfooders, re-release, real-time, reality-brillen, reality-programma, reality-soap, reality-toepassing, realitytelevisiereeksen, rechtvaardigheidsdiscours, record-knop, regiospinners, reisvriendelijke, robot-app, roboticamachines, rock-'n-rollfactor, rodeloperoutfits, rouwapplaus, rusthuisbevaks, salafi-jihadisten, salami-politiek, schijn-zzp'ers, schoolpoortmama's, schuldenbank, sci-fi, sciencefictionvideogame, scifi-strips, selfie-doden, selfie-gerelateerde, selfie-relletje, selfiedoden, selfiemakende, selfienemers, selfsticks, sensor-interfaces, shariaraadsleden, shop-and-go-parkeerplaatsen, shopfaciliteiten, show-events, singer-songwritermuziek, sjoemelsoftwarevrije, sjoemeltaks, sjoemeltoestellen, sjoemeltreinen, sjoemelvlees, skatestoppers, slapscreentjes, sludgecorehelden, sluikstort-hotspots, sluikstortapp, smartphone-ambities, smartphone-camera, smartphone-edities, smartphone-parkeren, smartphone-topmodellen, smartphonebeelden, smartphonecomponenten, smartphonefrontjes, smartphonekijkers, smartphonelichtjes, smartphoneontwikkelaars, smartwatch'-markt, smiley-eclips, snelle-interventiebrigade, snorcooters, socialemediapagina's, soepwijven, software-ingenieur, sorry-zo-was-het-niet-bedoeld'-teken, soundstickers, spaarplafond-formules, spaarvijvers, spaghetti-arrest, spam-mails, speeddate-baby's,

speeddating-sessies, spellingstoornis, spookbieders, staartstroppen, stadionsynths, stadsscouts, stadstuininitiatieven, stand-alone, start-up-community, start-up-steden, start-upregio, state-of-the-artvoorzieningen, steamingdiensten, stekkerbussen, steunzorgen, stiff-upper-lipkranten, stockcontracten, streaming-gegevens, streaming-licenties, streamingaanbieders, streamingcharts, streamingklanten, street-art-werken, stress-stoornis, suggestiefietsroutes, surround-installaties, symbooltrofeeën, sympatico's, tablet-apps, tablet-ontwikkelaars, tabletkijkers, tape-loops, taxi-oorlog, taxshiftonderhandelingen, teamteachers, techoplossingen, techprojecten, tegenwindfietsen, tennissneakers, terreurduo's, terroristenduo's, testdrones, testosterongames, thuisboxen, tienerblogs, tirolerfilms, too-big-to-fail-banken, topstreamers, totaal-ervaring, touch-screens, trackingbedrijven, transgendercafés, transmannenclubs, trashmetalpunks, treinplassen, treinwateren, trek-pleisters, trojka-keuzes, turbokleuters, turteltaks, tv-kookboeken, tv-on-demand-producten, tv-op-aanvraag-abonnementen, twintigerhipsters, underdog'-positie, undergroundhipsters, urine-detectors, vaatbommen, vacuümurinoirs, varkensweitjes, verstrip, vetloopjes, viager'-beleggingsfonds, viager'-systeem, video-essay, video-on-demand, videostreamingspelers, vierentwintiguursapotheek, vierverdieners, vintagemeubels, visverlaters, vluchtelingen-kampen, vluchtelingen'crisis, vluchtelingencoördinator, voedingspsychologe, voedselbankgebruikers, voedselmobieltjes, voedselveiligheidsagentschappen, vogelherkennings-apps, voxpop-formats, vr-bril, vreetmobieltje, vreugde'-emoji, vriend-met-voordelen, vrouwenechtparen, vte's, waardenpolitiek, wajong-uitkering, wallstickers, webcam-les, webvideo's, webwinkelactiviteiten, wegwijsvragen, wellness-keten, wespenpleisters, wietstekjes, wifibussen, wifiplekken, wifitrans, wijkondersteunende, wildlife-crime, woekeroksels, woonzorgkamers, woordgravers, x-rated, yezidi-mannen, yogataaltje, yogawinkeltje, yoghurtbommetjes, you'-syndroom, zaydi's, zeewiersmoothies, zelfmoordpropagandeurs, zero-sum, zero-taks, zorgtech-bedrijven, zpp'ers, zwerfvuilverpakkingen, zzp-jongeren, zzp'ende

Appendix III: Pythonscripts

```
# libraries importeren
import pandas as pd
import requests
import json
import xmltodict
from io import StringIO
from lxml import etree
from pathlib import Path
import sys

# lemmalijs WNW ophalen
lemmalijst =
requests.get("https://neologismen.ivdnt.org/backend/lemmalist?output=json")
lemmalijst.json()

# lemmalijs in een dataframe stoppen
neoDf = pd.DataFrame.from_records(lemmalijst.json())
neoDf.columns = ["lemma",
"lemmasamenvatting", "afbeelding", "video", "geluid", "kopjeOpnoemer"]
print(neoDf.describe())
neoDf.head()

# # artikel in XML van server ophalen en doorzoeken
#voorbeeldje kamelenkuch
response =
requests.get("https://neologismen.ivdnt.org/article/kamelenkuch?output=xml")
print(response.text)

#xml omzetten naar doorzoekbare tree
tree = etree.parse(StringIO(response.text))
# we zoeken naar xml-nodes met de naam VroegsteDatering
```

```

datering = tree.xpath('./VroegsteDatering')
print(len(datering))
print(datering[0].text)

# we zoeken naar xml-nodes met synoniemen en printen het aantal en de lijst
synoniemen = tree.xpath('./Synoniem/link/lemma')
print(len(synoniemen))
print(tree.xpath('./Synoniem/link/lemma/text()'))

# # Vroegste datering aan dataframe toevoegen

#Extra kolom voor Vroegste datering met default-waarde na (not available)
neoDf["Vroegst"] = "na"
neoDf["Datering_aantal"] = "na"

#voor elk lemma in dataframe wordt het artikel in XML opgehaald en gezocht op
VroegsteDatering
for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")
        tree = etree.parse(StringIO(response.text))
        datering = tree.xpath('./VroegsteDatering')
        if len(datering) > 0:
            neoDf.at[i,'Vroegst'] = datering[0].text
            neoDf.at[i,'Datering_aantal'] = len(datering)
    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

```

```
# # Woordsoort toevoegen
```

```
neoDf["Woordsoort"] = "na"
```

```
neoDf["Woordsoort_aantal"] = "na"
```

```
for i, row in neoDf.iterrows():
```

```
    try:
```

```
        response =
```

```
requests.get("https://neologismen.ivdnt.org/article/" + neoDf.at[i, 'lemma'] + "?output=xml")
```

```
        tree = etree.parse(StringIO(response.text))
```

```
        woordsoort = tree.xpath('.//Woordsoort/Type')
```

```
        if len(woordsoort) > 0:
```

```
            neoDf.at[i, 'Woordsoort'] = woordsoort[0].text
```

```
            neoDf.at[i, 'Woordsoort_aantal'] = len(woordsoort)
```

```
    except :
```

```
        print('no response for:' + neoDf.at[i, 'lemma'] )
```

```
neoDf.head()
```

```
neoDf['Woordsoort'].value_counts().plot(kind='barh')
```

```
# # Herkomst toevoegen
```

```
neoDf["Herkomst"] = "na"
```

```
neoDf["Herkomst_aantal"] = "na"
```

```
for i, row in neoDf.iterrows():
```

```
    try:
```

```
        response =
```

```
requests.get("https://neologismen.ivdnt.org/article/" + neoDf.at[i, 'lemma'] + "?output=xml")
```

```
        tree = etree.parse(StringIO(response.text))
```

```
        herkomst = tree.xpath('.//Etymologie/AardHerkomst')
```

```
        if len(herkomst) > 0:
```

```

neoDf.at[i,'Herkomst'] = herkomst[0].text
neoDf.at[i,'Herkomst_aantal'] = len(herkomst)
except :
    print('no response for:'+neoDf.at[i,'lemma'] )
count_herkomst = neoDf.pivot_table(columns=['Herkomst'], aggfunc='size')
tabel_herkomst = count_herkomst.sort_values(ascending=False)
print(tabel_herkomst)
pd.options.display.max_rows = 500

neoDf['Herkomst'].value_counts().plot(kind='barh')

## Brontaal toevoegen

neoDf["Brontaal"] = "na"
neoDf["Brontaal_aantal"] = "na"
for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xm
l")
        tree = etree.parse(StringIO(response.text))
        brontaal = tree.xpath('.//Ontlening/Brontaal')
        if len(brontaal) > 0:
            neoDf.at[i,'Brontaal'] = brontaal[0].text
            neoDf.at[i,'Brontaal_aantal'] = len(brontaal)
    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

## Woordvorming toevoegen

neoDf["Woordvorming"] = "na"
neoDf["Woordvorming_aantal"] = "na"

```



```

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+ "?output=xml")
        tree = etree.parse(StringIO(response.text))
        woordvorming = tree.xpath('./Woordvorming/WoordvormingType')
        if len(woordvorming) > 0:
            neoDf.at[i,'Woordvorming'] = woordvorming[0].text
            neoDf.at[i,'Woordvorming_aantal'] = len(woordvorming)
    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

## Linkerlid en rechterlid samenstelling toevoegen
neoDf["Linkerlid"] = "na"
neoDf["Rechterlid"] = "na"

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+ "?output=xml")
        tree = etree.parse(StringIO(response.text))
        linkerlid = tree.xpath('./Samenstelling/Linkerlid')
        if len(linkerlid) > 0:
            neoDf.at[i,'Linkerlid'] = linkerlid[0].text

    except :
        print('no response for:'+neoDf.at[i,'lemma'] )
for i, row in neoDf.iterrows():
    try:

```

```

response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")

tree = etree.parse(StringIO(response.text))
rechterlid = tree.xpath('./Samenstelling/Rechterlid')
if len(rechterlid) > 0:
    neoDf.at[i,'Rechterlid'] = rechterlid[0].text

except :
    print('no response for:'+neoDf.at[i,'lemma'] )

## Samenstellingen met trefwoord als linker/rechterlid toevoegen
neoDf["SamenstellingenMetTrefwoordAlsLinkerlid"] = "na"
neoDf["SamenstellingenMetTrefwoordAlsLinkerlid_aantal"] = "na"

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")

        tree = etree.parse(StringIO(response.text))
        trefwoordalslinkerlid = tree.xpath('./SamenstellingenMetTrefwoordAlsLinkerlid')
        if len(trefwoordalslinkerlid) > 0:
            neoDf.at[i,'SamenstellingenMetTrefwoordAlsLinkerlid'] =
trefwoordalslinkerlid[0].text
            neoDf.at[i,'SamenstellingenMetTrefwoordAlsLinkerlid_aantal'] =
len(trefwoordalslinkerlid)
        except :
            print('no response for:'+neoDf.at[i,'lemma'] )

neoDf["SamenstellingenMetTrefwoordAlsRechterlid"] = "na"
neoDf["SamenstellingenMetTrefwoordAlsRechterlid_aantal"] = "na"

```

```

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")
        tree = etree.parse(StringIO(response.text))
        trefwoordalsrechterlid =
tree.xpath('./SamenstellingenMetTrefwoordAlsRechterlid')
        if len(trefwoordalsrechterlid) > 0:
            neoDf.at[i,'SamenstellingenMetTrefwoordAlsRechterlid'] =
trefwoordalsrechterlid[0].text
            neoDf.at[i,'SamenstellingenMetTrefwoordAlsRechterlid_aantal'] =
len(trefwoordalsrechterlid)
        except :
            print('no response for:'+neoDf.at[i,'lemma'] )

# # Basisvorm en achtervoegsel afleiding toevoegen
neoDf["Basisvorm"] = "na"
neoDf["Achtervoegsel"] = "na"
for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")
        tree = etree.parse(StringIO(response.text))
        basisvorm = tree.xpath('./Afleiding/Basisvorm')
        if len(basisvorm) > 0:
            neoDf.at[i,'Basisvorm'] = basisvorm[0].text

    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

for i, row in neoDf.iterrows():

```

```

try:
    response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xm
l")
    tree = etree.parse(StringIO(response.text))
    achtervoegsel = tree.xpath('.//Afleiding/Achtervoegsel')
    if len(achtervoegsel) > 0:
        neoDf.at[i,'Achtervoegsel'] = achtervoegsel[0].text

except :
    print('no response for:'+neoDf.at[i,'lemma'] )

# # Domein toevoegen
neoDf["Domein"] = "na"
neoDf["Domein_aantal"] = "na"
for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xm
l")
        tree = etree.parse(StringIO(response.text))
        domein = tree.xpath('.//domeinBody/Koepeldomein')
        if len(domein) > 0:
            neoDf.at[i,'Domein'] = domein[0].text
            neoDf.at[i,'Domein_aantal'] = len(domein)
    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

# # Bedenker toevoegen

neoDf["Bedenker"] = "na"
neoDf["Bedenker_aantal"] = "na"

```

```

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+ "?output=xml")
        tree = etree.parse(StringIO(response.text))
        bedenker = tree.xpath('//*[Etymologie/Bedenker]')
        if len(bedenker) > 0:
            neoDf.at[i,'Bedenker'] = bedenker[0].text
            neoDf.at[i,'Bedenker_aantal'] = len(bedenker)
    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

```

Taalvarieteit toevoegen

```
neoDf["Taalvarieteit"] = "na"
```

```
neoDf["Taalvarieteit_aantal"] = "na"
```

```
for i, row in neoDf.iterrows():
```

```
    try:
```

```
        response =
```

```
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+ "?output=xml")
```

```
        tree = etree.parse(StringIO(response.text))
```

```
        taalvarieteit = tree.xpath('//*[BijzonderhedenGebruik/Taalvarieteit]')
```

```
        if len(taalvarieteit) > 0:
```

```
            neoDf.at[i,'Taalvarieteit'] = taalvarieteit[0].text
```

```
            neoDf.at[i,'Taalvarieteit_aantal'] = len(taalvarieteit)
```

```
    except :
```

```
        print('no response for:'+neoDf.at[i,'lemma'] )
```

Definitie toevoegen

```
neoDf["Definitie"] = "na"
```

```
neoDf["Definitie_aantal"] = "na"
```

```

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xm
l")
        tree = etree.parse(StringIO(response.text))
        definitie = tree.xpath('.//definitieBody/Definitie')
        if len(definitie) > 0:
            neoDf.at[i,'Definitie'] = definitie[0].text
            neoDf.at[i,'Definitie_aantal'] = len(definitie)
    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

# # Type linkerlid/rechterlid samenstelling toevoegen

```

```

neoDf["Linkerlidtype"] = "na"
neoDf["Rechterlidtype"] = "na"

```

```

for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xm
l")
        tree = etree.parse(StringIO(response.text))
        linkerlidtype = tree.xpath('.//Samenstelling/Linkerlidtype')
        if len(linkerlidtype) > 0:
            neoDf.at[i,'Linkerlidtype'] = linkerlidtype[0].text

    except :
        print('no response for:'+neoDf.at[i,'lemma'] )

```

```

for i, row in neoDf.iterrows():
    try:

```

```

        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")

        tree = etree.parse(StringIO(response.text))
        rechterlidtype = tree.xpath('//*[Samenstelling/Rechterlidtype]')
        if len(rechterlidtype) > 0:
            neoDf.at[i,'Rechterlidtype'] = rechterlidtype[0].text

except :
    print('no response for:'+neoDf.at[i,'lemma'] )

## Houding toevoegen
neoDf["Houding"] = "na"
for i, row in neoDf.iterrows():
    try:
        response =
requests.get("https://neologismen.ivdnt.org/article/"+neoDf.at[i,'lemma']+"?output=xml")

        tree = etree.parse(StringIO(response.text))
        houding = tree.xpath('//*[BijzonderhedenGebruik/Houding]')
        if len(houding) > 0:
            neoDf.at[i,'Houding'] = houding[0].text

except :
    print('no response for:'+neoDf.at[i,'lemma'] )

#geslacht en lidwoord en synoniem toevoegen
wnw["Geslacht"] = "na"
wnw["Lidwoord"] = "na"

for i, row in wnw.iterrows():
    try:

```

```

        response =
requests.get("https://neologismen.ivdnt.org/article/"+wnw.at[i,'lemma']+ "?output=xml"
)
        tree = etree.parse(StringIO(response.text))
        geslacht = tree.xpath('.//Geslacht')
        if len(geslacht) > 0:
            wnw.at[i,'Geslacht'] = geslacht[0].text
except :
    print('no response for:'+wnw.at[i,'lemma'] )

```

```

for i, row in wnw.iterrows():

```

```

    try:

```

```

        response =
requests.get("https://neologismen.ivdnt.org/article/"+wnw.at[i,'lemma']+ "?output=xml"
)
        tree = etree.parse(StringIO(response.text))
        lidwoord = tree.xpath('.//AardLidwoord')
        if len(lidwoord) > 0:
            wnw.at[i,'Lidwoord'] = lidwoord[0].text
except :
    print('no response for:'+wnw.at[i,'lemma'] )

```

```

wnw["Synoniem"] = "na"

```

```

for i, row in wnw.iterrows():

```

```

    try:

```

```

        response =
requests.get("https://neologismen.ivdnt.org/article/"+wnw.at[i,'lemma']+ "?output=xml"
)
        tree = etree.parse(StringIO(response.text))
        synoniem = tree.xpath('.//Synoniem/link/lemma')
        if len(synoniem) > 0:
            wnw.at[i,'Synoniem'] = len(synoniem)

```



```

except :
    print('no response for:'+wnw.at[i,'lemma'] )

# # Wegschrijven pandas dataframe naar csv file ANW
#wegschrijven naar file anwNeo.txt
with Path('anwNeo.txt').open('w') as f:
    neoDf.to_csv(f, sep='\t', index=False, line_terminator='\n')

#productiviteit van linker- en rechterlid
wnw =
pd.read_csv(Path('/media/projecten/Onderwijs/Masterproeven/VanLintElla/WNW+fre
quentieverloop.txt'),
            encoding = "ISO-8859-1", header = 0, sep= '\t')
wnw.head()

url = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22BN%22+%22NN%22%29&first=0'+
'&group=hit%3Alemma%3Ai&number=20&patt=%5B+word+%3D+%22bakfiets...%2
B%22+%26'+ ' '+word+%21%3D+%22bakfietsen%22+%26'+
'+word+%21%3D+%22bakfietsen%22+%5D'+ '&sort=identity'+
'&outputformat=json'

f = urllib.request.urlopen(url)
response = json.loads(f.read().decode('utf-8'))
response

wnw["aantal_samenstelling_LL"] = "na"
wnw["cumulfreq_LL"] = "na"
wnw["aantal_samenstelling_RL"] = "na"
wnw["cumulfreq_RL"] = "na"

```

```
# # Linkerlid
```

```
for i, row in wnw.iterrows():
```

```
    try:
```

```
        lemma = wnw.at[i, 'lemma']
```

```
        linkerlid = wnw.at[i, 'Linkerlid']
```

```
        woordsoort = wnw.at[i, 'Woordsoort']
```

```
        if linkerlid != "na" and woordsoort != "werkwoord (stam)" and woordsoort !=  
"werkwoord (infinitief)":
```

```
            url = 'http://chn-i.ivdnt.loc/blacklab-server/chn-  
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+  
'+newspapers_languageVariant%3A%28%22BN%22+%22NN%22%29&first=0'+  
'&group=hit%3Alemma%3Ai&number=1000&patt=' +  
'%5B+word+%3D+%22'+urllib.parse.quote_plus(linkerlid)+  
'...%2B%22+%26+word+%21%3D+%22'+urllib.parse.quote_plus(lemma)+  
'%22+%5D'+          '&sort=identity'+          '&outputformat=json'
```

```
            f = urllib.request.urlopen(url)
```

```
            response = json.loads(f.read().decode('utf-8'))
```

```
            df = pd.json_normalize(response['hitGroups'])
```

```
            access = response['summary']
```

```
            groups = access['numberOfGroups']
```

```
            wnw.at[i, 'aantal_samenstelling_LL'] = groups
```

```
            size = access['numberOfHits']
```

```
            wnw.at[i, 'cumulfreq_LL'] = size
```

```
    except:
```

```
        print('no response for:'+wnw.at[i, 'lemma'])
```

```
# # Linkerlid werkwoord
```

```
for i, row in wnw.iterrows():
```

```
    try:
```

```
        lemma = wnw.at[i, 'lemma']
```

```
        woordsoort = wnw.at[i, 'Woordsoort']
```

```
        stam = wnw.at[i, 'Stam']
```

```
        if linkerlid != "na":
```

```
            if woordsoort == "werkwoord (stam)" or woordsoort == "werkwoord  
(infinitief)":
```

```
                url = 'http://chn-i.ivdnt.loc/blacklab-server/chn-  
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+  
'+newspapers_languageVariant%3A%28%22BN%22+%22NN%22%29&first=0'+  
'&group=hit%3Alemma%3Ai&number=1000&patt=' +  
'%5B+word+%3D+%22'+urllib.parse.quote_plus(stam)+  
'...%2B%22+%26+word+%21%3D+%22'+urllib.parse.quote_plus(lemma)+  
'%22+%5D'+                '&sort=identity'+                '&outputformat=json'
```

```
                    f = urllib.request.urlopen(url)
```

```
                    response = json.loads(f.read().decode('utf-8'))
```

```
                    df = pd.json_normalize(response['hitGroups'])
```

```
                    access = response['summary']
```

```
                    groups = access['numberOfGroups']
```

```
                    wnw.at[i, 'aantal_samenstelling_LL'] = groups
```

```
                    size = access['numberOfHits']
```

```
                    wnw.at[i, 'cumulfreq_LL'] = size
```

```
    except:
```

```
        print('no response for:'+wnw.at[i, 'lemma'])
```

```

# # Rechterlid
for i, row in wnw.iterrows():
    try:
        lemma = wnw.at[i, 'lemma']
        rechterlid = wnw.at[i, 'Rechterlid']
        if rechterlid != "na":
            url = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22BN%22+%22NN%22%29&first=0'+
'&group=hit%3Alemma%3Ai&number=1000&patt=' +
'%5B+word+%3D+%22.%2A'+urllib.parse.quote_plus(rechterlid)+
'%22+%26+word+%21%3D+%22'+urllib.parse.quote_plus(rechterlid)+
'%22+%26+word+%21%3D+%22'+urllib.parse.quote_plus(lemma)+
'%22+%5D'+          '&sort=identity'+          '&outputformat=json'

            f = urllib.request.urlopen(url)
            response = json.loads(f.read().decode('utf-8'))
            df = pd.json_normalize(response['hitGroups'])

            access = response['summary']

            groups = access['numberOfGroups']
            wnw.at[i, 'aantal_samenstelling_RL'] = groups

            size = access['numberOfHits']
            wnw.at[i, 'cumulfreq_RL'] = size
    except:
        print('no response for:'+wnw.at[i, 'lemma'])

with
Path('/media/projecten/Onderwijs/Masterproeven/VanLintElla/productiviteit.txt').open(
'w') as f:
    wnw.to_csv(f, sep='\t', index=False, line_terminator='\n')

```

```

# # Inlezen van tab-separated txt file Neoloog
neoloog = pd.read_csv(Path('neoloog_data-1636019112416_deduplicated.txt'), sep=
\t', index_col = 0)
neoloog.head()

# # Neoloog en ANW mergen
neoloog_neoDf = neoloog.merge(neoDf, on="lemma", how="outer")
neoloog_neoDf.head()
with Path('neoloog_neoDf_outer.txt').open('w') as f:
    neoloog_neoDf.to_csv(f, sep='\t', index=False, line_terminator='\n')

# Levenshteinafstand berekenen
#eerst installeren we de Levenshtein module
import sys
get_ipython().system('{sys.executable} -m pip install python-Levenshtein')
import Levenshtein as ls

#De functie van Keuleers om vergelijkwoorden te sorteren op lengte

def maketargets(itemlist):
    """Make a dictionary of length indexed strings.

    itemlist: a list of strings"""
    targets={}
    for item in itemlist:
        targets.setdefault(len(item), []).append(item)
    return targets

#De functie van Keuleers om efficient Levenshteinafstanden te bereken, maar nu
geeft hij de meest gelijkende woorden terug

def kdistribution(source, targets, minimum=0, k=None):
    """Compute the distribution of Levenshtein distances given a source string

```

and a dictionary with length-indexed target strings (see maketargets). This version orders computation of distances by absolute difference in length between source and target.

When a value is given for the argument k, computation finishes when k neighbors have been found at a distance that is smaller than the minimal absolute difference in length of the remaining targets. This considerably speeds up giving limited distributions for k neighbors when k is relatively low, and is a very fast method of finding the distance of the neighbors at the nearest distance.

Arguments

source: a source string

targets: a length-indexed dictionary of targets (see maketargets)

minimum: the minimum Levenshtein distance to include in the distribution. Default is 0, but should be set to 1 if counting identical neighbors is not desired.

k: the minimal number of neighbors to make a distribution for.

"""

```
distribution={}
knearest ={}
sourcelen=len(source)
order=sorted(targets, key=lambda x: abs(sourcelen-x))
previous_difference=0
for targetlen in order:
    difference=(abs(sourcelen-targetlen))
    if difference > previous_difference:
        kmin=sum([distribution.get(distance,0) for distance in range(minimum,
difference)])
        #kmin=sum([len(knearest.get(distance,[])) for distance in range(minimum,
difference)])
        if k!=None and kmin>k:
```

```

        break
    else:
        previous_difference=difference
    for target in targets[targetlen]:
        distance=ls.distance(source, target)
        distribution[distance]=distribution.get(distance,0)+1
        knearest[distance]=knearest.get(distance,[])+[target]
    for i in range(0, minimum):
        if i in distribution: del distribution[i]
    return knearest

import pandas as pd
# we lezen de WNW dataset in (een versie in UTF8 met separator ',')
wnw = pd.read_csv('wnw+neo.csv', sep=';', header=0)

# we maken een lijst van alle lemmata in de wnw dataset
neos = wnw['lemma'].tolist()
neos

# we lezen de SUBTLEX frequentie dataset in
SUBTLEX = pd.read_csv('SUBTLEX-NL.cd-above2.tsv', sep='\t', header=0)
SUBTLEX

# we nemen de 10000 meest frequente woorden als vergelijkingsset
lexicon = SUBTLEX['Word'].head(10000).tolist()
print(lexicon[0],lexicon[-1])

# we maken van het lexicon een op lengte gesorteerde dictionary met Keuleers'
maketargets()
targets=maketargets(lexicon)

# we halen met Keuleers kdistribution() de minimaal 3 meest gelijkende woorden uit
SUBTLEX voor elke neologisme

```

```

result={}
for neo in neos:
    result[neo] = kdistribution(neo, targets, minimum=1, k=3)

# in de wnw dataset maken we kolommen bij om naast de Levenshtein afstand zelf,
ook
wnw['LevenshteinMinWord'] = 'na' # het meest frequente, meest gelijkende woord
wnw['LevenshteinMinWordFreq'] = 'na' # de frequentie van dat woord (volgens
SUBTLEX)
wnw['LevenshteinMinWordLogFreq'] = 'na' # de log10 frequentie van het woord

# uit de dictionary met resultaten en SUBTLEX plukken we per neologisme de
relevante info in stoppen hem in het wnw dataframe

for neo in result:
    wnw.loc[wnw['lemma']==neo,['Levenshtein']] = min(result[neo].keys())
    wnw.loc[wnw['lemma']==neo,['LevenshteinMinWord']] =
SUBTLEX[SUBTLEX['Word'].isin(result[neo][min(result[neo].keys())]).sort_values('F
REQcount').iloc[-1]]['Word']
    wnw.loc[wnw['lemma']==neo,['LevenshteinMinWordFreq']] =
SUBTLEX[SUBTLEX['Word'].isin(result[neo][min(result[neo].keys())]).sort_values('F
REQcount').iloc[-1]]['FREQcount']
    wnw.loc[wnw['lemma']==neo,['LevenshteinMinWordLogFreq']] =
SUBTLEX[SUBTLEX['Word'].isin(result[neo][min(result[neo].keys())]).sort_values('F
REQcount').iloc[-1]]['Lg10WF']

# ter illustratie voor "aanblijfrpemie"
wnw[wnw['lemma']=='aanblijfrpemie'][['lemma','Levenshtein','LevenshteinMinWord','L
evenshteinMinWordFreq','LevenshteinMinWordLogFreq']]

# even kijken in het wnw dataframe wat de meest en minst gelijkende woorden
waren

```



```
wnw.sort_values('Levenshtein')[['lemma','Levenshtein','LevenshteinMinWord','LevenshteinMinWordFreq','LevenshteinMinWordLogFreq']]
```

```
# we schrijven het wnw dataframe weg als tsv
```

```
from pathlib import Path
```

```
with Path('wnw_df.txt').open('w') as f:
```

```
    wnw.to_csv(f, sep='\t', index=False, line_terminator='\n')
```

```
# frequentieverloop wnw
```

```
import pandas as pd
```

```
import urllib.parse
```

```
import urllib.request
```

```
import requests
```

```
import json
```

```
import xmltodict
```

```
from io import StringIO
```

```
from lxml import etree
```

```
from pathlib import Path
```

```
import sys
```

```
import re
```

```
wnw =
```

```
pd.read_csv(Path('/media/projecten/Onderwijs/Masterproeven/VanLintElla/frequentieverloopWNW.txt'),
```

```
            sep= '\t')
```

```
# # Kolommen toevoegen
```

```
wnw["chn_vroegste_datering"] = "na"
```

```
wnw["chn_vroegste_datering_varieteit"] = "na"
```

```
wnw["BN_0"] = "na"
```

```
wnw["NN_0"] = "na"
```

```
wnw["BN_L"] = "na"
```

```

wnw["NN_L"] = "na"

def get_year(inputDic, outputList):
    for year in inputDic:
        year_access = year["identityDisplay"]
        outputList.append(int(year_access[0:4]))

for i, row in wnw.iterrows():
    try:
        lemma = wnw.at[i, 'lemma']
        if re.search(' ', lemma):
            searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"]][lemma="'+re.sub(' ','-',lemma)+'"]]'
        else:
            searchpattern = "'"+lemma+"'"
        url_BN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22BN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+'          '&sort=identity'+
'&outputformat=json'

        f_BN = urllib.request.urlopen(url_BN)
        response_BN = json.loads(f_BN.read().decode('utf-8'))
        df_BN = pd.json_normalize(response_BN['hitGroups'])

        save_year = []

        get_year(response_BN['hitGroups'], save_year)
        if len(save_year) > 0:
            wnw.at[i, "BN_0"] = save_year[0]
    except:
        print('no response for:'+wnw.at[i,'lemma'])

```

```

for i, row in wnw.iterrows():
    try:
        lemma = wnw.at[i, 'lemma']
        if re.search(' ', lemma):
            searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"]][lemma="'+re.sub(' ','-',lemma)+'"]]'
        else:
            searchpattern = '"'+lemma+'"'
        url_NN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22NN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+'          '&sort=identity'+
'&outputformat=json'

        f_NN = urllib.request.urlopen(url_NN)
        response_NN = json.loads(f_NN.read().decode('utf-8'))
        df_NN = pd.json_normalize(response_NN['hitGroups'])

        save_year = []

        get_year(response_NN['hitGroups'], save_year)
        if len(save_year) > 0:
            wnw.at[i, "NN_0"] = save_year[0]

    except:
        print('no response for:'+wnw.at[i,'lemma'])

for i, row in wnw.iterrows():
    try:
        lemma = wnw.at[i, 'lemma']
        if re.search(' ', lemma):

```

```

        searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"]][lemma="'+re.sub(' ','-',lemma)+'"])'
    else:
        searchpattern = '"+lemma+"'
    url_BN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22BN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+'&sort=identity'+
'&outputformat=json'

    f_BN = urllib.request.urlopen(url_BN)
    response_BN = json.loads(f_BN.read().decode('utf-8'))
    df_BN = pd.json_normalize(response_BN['hitGroups'])

    save_year = []

    get_year(response_BN['hitGroups'], save_year)
    if len(save_year) > 0:
        wnw.at[i, "BN_L"] = save_year[-1]
    except:
        print('no response for:'+wnw.at[i,'lemma'])

for i, row in wnw.iterrows():
    try:
        lemma = wnw.at[i, 'lemma']
        if re.search(' ', lemma):
            searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"]][lemma="'+re.sub(' ','-',lemma)+'"])'
        else:
            searchpattern = '"+lemma+"'
        url_NN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+

```

```
'+newspapers_languageVariant%3A%28%22NN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+'&sort=identity'+
'&outputformat=json'
```

```
f_NN = urllib.request.urlopen(url_NN)
response_NN = json.loads(f_NN.read().decode('utf-8'))
df_NN = pd.json_normalize(response_NN['hitGroups'])
```

```
save_year = []
```

```
get_year(response_NN['hitGroups'], save_year)
if len(save_year) > 0:
    wnw.at[i, "NN_L"] = save_year[-1]
```

```
except:
```

```
print('no response for:'+wnw.at[i,'lemma'])
```

```
# # Vroegste datering corpus toevoegen
```

```
for i, row in wnw.iterrows():
```

```
try:
```

```
    datering_NN = int(re.sub('na', '9999',str(wnw.at[i, "NN_0"])))
    datering_BN = int(re.sub('na', '9999',str(wnw.at[i, "BN_0"])))
    vroegste_datering_corpus = min(datering_NN, datering_BN)
    if vroegste_datering_corpus != 9999:
        wnw.at[i, "chn_vroegste_datering"] = vroegste_datering_corpus
    if datering_NN != 9999 and datering_BN != 9999:
        if datering_NN > datering_BN:
            wnw.at[i, "chn_vroegste_datering_varieteit"] = "BN"
        if datering_NN < datering_BN:
            wnw.at[i, "chn_vroegste_datering_varieteit"] = "NN"
```

```

    if datering_NN == datering_BN:
        wnw.at[i, "chn_vroegste_datering_varieteit"] = "BN+NN"
    if datering_NN == 9999 and datering_BN != 9999:
        wnw.at[i, "chn_vroegste_datering_varieteit"] = "BN"
    if datering_NN != 9999 and datering_BN == 9999:
        wnw.at[i, "chn_vroegste_datering_varieteit"] = "NN"
except:
    print('no response for:'+wnw.at[i,'lemma'])

wnw= wnw.reindex(columns=(wnw.columns.tolist() + [s+'BNfreq' for s in
list(map(str,range(1999,2022)))]
        + [s+'BNsubcorpSize' for s in list(map(str,range(1999,2022)))] +
        [s+'BNdocs' for s in list(map(str,range(1999,2022)))] +
        [s+'NNfreq' for s in list(map(str,range(1999,2022)))] +
        [s+'NNsubcorpSize' for s in list(map(str,range(1999,2022)))] +
        [s+'NNdocs' for s in list(map(str,range(1999,2022)))]), fill_value='na')
wnw.head()

# Frequentieverloop NN

languageVariant = "NN"
for i, row in wnw.iterrows():
    try:
        lemma = wnw.at[i, 'lemma']
        if re.search(' ', lemma):
            searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"])[lemma="'+re.sub(' ','-',lemma)+'"]]'
        else:
            searchpattern = '"'+lemma+'"'
        url= 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22'+languageVariant +'%22%29&first=0'+

```

```

'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+
'&sort=identity'+
'&outputformat=json'
    f = urllib.request.urlopen(url)
    response = json.loads(f.read().decode('utf-8'))
    df = pd.json_normalize(response['hitGroups'])
    if len(df) > 0:
        df['year'] = [int(x[0:4]) for x in df['identityDisplay'].tolist()] #maak de kolom
'year' als integer van de 4 eerste cijfers uit identityDisplay
        wnw.at[i,languageVariant + '_0'] = df['year'][0]
        for year in range(1999,2022):
            if year in df['year'].tolist():
                wnw.at[i,str(year)+languageVariant +'freq'] =
df[df['year']==year]['size'].tolist()[0]
                wnw.at[i,str(year)+languageVariant +'docs'] =
df[df['year']==year]['numberOfDocs'].tolist()[0]
                wnw.at[i,str(year)+languageVariant +'subcorpSize'] =
df[df['year']==year]['subcorpusSize.tokens'].tolist()[0]
            else:
                wnw.at[i,str(year)+languageVariant +'freq'] = 0
                wnw.at[i,str(year)+languageVariant +'docs'] = 0
                wnw.at[i,str(year)+languageVariant +'subcorpSize'] = 0
        except:
            print('no response for: '+wnw.at[i,'lemma'])

wnw[["lemma", "NN_0", "2019NNfreq"]].head()

# Frequentieverloop BN

languageVariant = "BN"
for i, row in wnw.iterrows():
    try:

```

```

lemma = wnw.at[i, 'lemma']
if re.search(' ', lemma):
    searchpattern = '([word="'+lemma.split(' ')[0]+'"][lemma="'+lemma.split('
')[1]+'"])[lemma="'+re.sub(' ','-',lemma)+'"]]'
else:
    searchpattern = '"+lemma+"'
url= 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22'+languageVariant +'%'22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&pat='+urllib.parse.quote_plus(searchpattern)+'&sort=identity'+
'&outputformat=json'
f = urllib.request.urlopen(url)
response = json.loads(f.read().decode('utf-8'))
df = pd.json_normalize(response['hitGroups'])
if len(df) > 0:
    df['year'] = [int(x[0:4]) for x in df['identityDisplay'].tolist()] #maak de kolom
'year' als integer van de 4 eerste cijfers uit identityDisplay
    wnw.at[i,languageVariant + '_0'] = df['year'][0]
    for year in range(1999,2022):
        if year in df['year'].tolist():
            wnw.at[i,str(year)+languageVariant +'freq'] =
df[df['year']==year]['size'].tolist()[0]
            wnw.at[i,str(year)+languageVariant +'docs'] =
df[df['year']==year]['numberOfDocs'].tolist()[0]
            wnw.at[i,str(year)+languageVariant +'subcorpSize'] =
df[df['year']==year]['subcorpusSize.tokens'].tolist()[0]
        else:
            wnw.at[i,str(year)+languageVariant +'freq'] = 0
            wnw.at[i,str(year)+languageVariant +'docs'] = 0
            wnw.at[i,str(year)+languageVariant +'subcorpSize'] = 0
except:
    print('no response for: '+wnw.at[i,'lemma'])

```



```

# # Wegschrijven

with
Path('/media/projecten/Onderwijs/Masterproeven/VanLintElla/leeftijdna2003.txt').open('w') as f:
    www.to_csv(f, sep='\t', index=False, line_terminator='\n')

# frequentieverloop neoloog

import pandas as pd
import urllib.parse
import urllib.request
import requests
import json
import xmltodict
from io import StringIO
from lxml import etree
from pathlib import Path
import sys
import re

neoloog =
pd.read_csv(Path('/media/projecten/Onderwijs/Masterproeven/VanLintElla/neoloog_data-1636019112416.txt'),
            sep= '\t', index_col = 0)
neoloog.head()

neo2015 =
neoloog[(neoloog.datum.str.match('2015'))].drop_duplicates(subset=["lemma"],keep='first')
neo2015.head(50
    )

```

```
# # Kolommen toevoegen
```

```
neo2015["chn_vroegste_datering"] = "na"  
neo2015["chn_vroegste_datering_varieteit"] = "na"  
neo2015["BN_0"] = "na"  
neo2015["NN_0"] = "na"  
neo2015["BN_L"] = "na"  
neo2015["NN_L"] = "na"
```

```
def get_year(inputDic, outputList):
```

```
    for year in inputDic:  
        year_access = year["identityDisplay"]  
        outputList.append(int(year_access[0:4]))
```

```
for i, row in neo2015.iterrows():
```

```
    try:  
        lemma = neo2015.at[i, 'lemma']  
        if re.search(' ', lemma):  
            searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('')[1]+'"][[lemma="'+re.sub(' ','-',lemma)+'"]]'  
        else:  
            searchpattern = "'"+lemma+"'"  
        url_BN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-  
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+  
'+newspapers_languageVariant%3A%28%22BN%22%29&first=0'+  
'&group=field%3Agrouping_year%3Ai&number=20'+  
'&patt='+urllib.parse.quote_plus(searchpattern)+'&sort=identity'+  
'&outputformat=json'  
  
        f_BN = urllib.request.urlopen(url_BN)  
        response_BN = json.loads(f_BN.read().decode('utf-8'))  
        df_BN = pd.json_normalize(response_BN['hitGroups'])
```

```

save_year = []

get_year(response_BN['hitGroups'], save_year)
if len(save_year) > 0:
    neo2015.at[i, "BN_0"] = save_year[0]
except:
    print('no response for:'+neo2015.at[i,'lemma'])

for i, row in neo2015.iterrows():
    try:
        lemma = neo2015.at[i, 'lemma']
        if re.search(' ', lemma):
            searchpattern = '([word="'+lemma.split(' ')[0]+'"]|[lemma="'+lemma.split('
')[1]+'"]|[lemma="'+re.sub(' ','-',lemma)+'"]]'
        else:
            searchpattern = '"'+lemma+'"'
        url_NN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22NN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'& patt='+urllib.parse.quote_plus(searchpattern)+'          '&sort=identity'+
'&outputformat=json'

        f_NN = urllib.request.urlopen(url_NN)
        response_NN = json.loads(f_NN.read().decode('utf-8'))
        df_NN = pd.json_normalize(response_NN['hitGroups'])

        save_year = []

        get_year(response_NN['hitGroups'], save_year)
        if len(save_year) > 0:
            neo2015.at[i, "NN_0"] = save_year[0]

```

```

except:
    print('no response for:'+neo2015.at[i,'lemma'])

for i, row in neo2015.iterrows():
    try:
        lemma = neo2015.at[i, 'lemma']
        if re.search(' ', lemma):
            searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"][[lemma="'+re.sub(' ','-',lemma)+'"]]'
        else:
            searchpattern = '"+lemma+"'
        url_BN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22BN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+'
'&sort=identity'+
'&outputformat=json'

        f_BN = urllib.request.urlopen(url_BN)
        response_BN = json.loads(f_BN.read().decode('utf-8'))
        df_BN = pd.json_normalize(response_BN['hitGroups'])

        save_year = []

        get_year(response_BN['hitGroups'], save_year)
        if len(save_year) > 0:
            neo2015.at[i, "BN_L"] = save_year[-1]
    except:
        print('no response for:'+neo2015.at[i,'lemma'])

for i, row in neo2015.iterrows():
    try:

```

```

lemma = neo2015.at[i, 'lemma']
if re.search(' ', lemma):
    searchpattern = '([word="'+lemma.split(' ')[0]+'"][lemma="'+lemma.split('
')[1]+'"])[lemma="'+re.sub(' ','-',lemma)+'"]'
else:
    searchpattern = '"+lemma+"'
url_NN = 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22NN%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&patt='+urllib.parse.quote_plus(searchpattern)+'          '&sort=identity'+
'&outputformat=json'

f_NN = urllib.request.urlopen(url_NN)
response_NN = json.loads(f_NN.read().decode('utf-8'))
df_NN = pd.json_normalize(response_NN['hitGroups'])

save_year = []

get_year(response_NN['hitGroups'], save_year)
if len(save_year) > 0:
    neo2015.at[i, "NN_L"] = save_year[-1]

except:
    print('no response for:'+neo2015.at[i,'lemma'])

# # Vroegste datering corpus toevoegen

for i, row in neo2015.iterrows():
    try:
        datering_NN = int(re.sub('na', '9999',str(neo2015.at[i, "NN_0"])))
        datering_BN = int(re.sub('na', '9999',str(neo2015.at[i, "BN_0"])))
        vroegste_datering_corpus = min(datering_NN, datering_BN)

```

```

if vroegste_datering_corpus != 9999:
    neo2015.at[i, "chn_vroegste_datering"] = vroegste_datering_corpus
if datering_NN != 9999 and datering_BN != 9999:
    if datering_NN > datering_BN:
        neo2015.at[i, "chn_vroegste_datering_varieteit"] = "BN"
    if datering_NN < datering_BN:
        neo2015.at[i, "chn_vroegste_datering_varieteit"] = "NN"
    if datering_NN == datering_BN:
        neo2015.at[i, "chn_vroegste_datering_varieteit"] = "BN+NN"
if datering_NN == 9999 and datering_BN != 9999:
    neo2015.at[i, "chn_vroegste_datering_varieteit"] = "BN"
if datering_NN != 9999 and datering_BN == 9999:
    neo2015.at[i, "chn_vroegste_datering_varieteit"] = "NN"
except:
    print('no response for:'+neo2015.at[i,'lemma'])

```

```

neo2015= neo2015.reindex(columns=(neo2015.columns.tolist() + [s+'BNfreq' for s in
list(map(str,range(1999,2022))]))
    + [s+'BNsubcorpSize' for s in list(map(str,range(1999,2022)))] +
    [s+'BNdocs' for s in list(map(str,range(1999,2022)))] +
    [s+'NNfreq' for s in list(map(str,range(1999,2022)))] +
    [s+'NNsubcorpSize' for s in list(map(str,range(1999,2022)))] +
    [s+'NNdocs' for s in list(map(str,range(1999,2022)))]), fill_value='na')
neo2015.head()

```

```

# # Frequentieverloop NN

```

```

languageVariant = "NN"
for i, row in neo2015.iterrows():
    try:
        lemma = neo2015.at[i, 'lemma']
        if re.search(' ', lemma):

```

```

        searchpattern = '([word="'+lemma.split(' ')[0]+'"])[lemma="'+lemma.split('
')[1]+'"])[lemma="'+re.sub(' ','-',lemma)+'"]'
    else:
        searchpattern = lemma
    url= 'http://chn-i.ivdnt.loc/blacklab-server/chn-
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+
'+newspapers_languageVariant%3A%28%22'+languageVariant +'%22%29&first=0'+
'&group=field%3Agrouping_year%3Ai&number=20'+
'&pat='+urllib.parse.quote_plus(searchpattern)+'&sort=identity'+
'&outputformat=json'
    f = urllib.request.urlopen(url)
    response = json.loads(f.read().decode('utf-8'))
    df = pd.json_normalize(response['hitGroups'])
    if len(df) > 0:
        df['year'] = [int(x[0:4]) for x in df['identityDisplay'].tolist()] #maak de kolom
'year' als integer van de 4 eerste cijfers uit identityDisplay
        neo2015.at[i,languageVariant+'_0'] = df['year'][0]
        for year in range(1999,2022):
            if year in df['year'].tolist():
                neo2015.at[i,str(year)+languageVariant+'freq'] =
df[df['year']==year]['size'].tolist()[0]
                neo2015.at[i,str(year)+languageVariant+'docs'] =
df[df['year']==year]['numberOfDocs'].tolist()[0]
                neo2015.at[i,str(year)+languageVariant+'subcorpSize'] =
df[df['year']==year]['subcorpusSize.tokens'].tolist()[0]
            else:
                neo2015.at[i,str(year)+languageVariant+'freq'] = 0
                neo2015.at[i,str(year)+languageVariant+'docs'] = 0
                neo2015.at[i,str(year)+languageVariant+'subcorpSize'] = 0
    except:
        print('no response for: '+neo2015.at[i,'lemma'])

```

```
# # Frequentieverloop BN
```

```
languageVariant = "BN"
```

```
for i, row in neo2015.iterrows():
```

```
    try:
```

```
        lemma = neo2015.at[i, 'lemma']
```

```
        if re.search(' ', lemma):
```

```
            searchpattern = '([word="'+lemma.split(' ')[0]+'"][lemma="'+lemma.split('')[1]+'"])[lemma="'+re.sub(' ','-',lemma)+'"]'
```

```
        else:
```

```
            searchpattern = '"+lemma+"'
```

```
            url= 'http://chn-i.ivdnt.loc/blacklab-server/chn-  
intern/hits?filter=medium%3A%28%22newspaper%22%29+AND'+  
'+newspapers_languageVariant%3A%28%22'+languageVariant +'%22%29&first=0'+  
'&group=field%3Agrouping_year%3Ai&number=20'+  
'&patt='+urllib.parse.quote_plus(searchpattern)+'&sort=identity'+  
'&outputformat=json'
```

```
            f = urllib.request.urlopen(url)
```

```
            response = json.loads(f.read().decode('utf-8'))
```

```
            df = pd.json_normalize(response['hitGroups'])
```

```
            if len(df) > 0:
```

```
                df['year'] = [int(x[0:4]) for x in df['identityDisplay'].tolist()] #maak de kolom  
'year' als integer van de 4 eerste cijfers uit identityDisplay
```

```
                neo2015.at[i,languageVariant + '_0'] = df['year'][0]
```

```
                for year in range(1999,2022):
```

```
                    if year in df['year'].tolist():
```

```
                        neo2015.at[i,str(year)+languageVariant +'freq'] =  
df[df['year']==year]['size'].tolist()[0]
```

```
                        neo2015.at[i,str(year)+languageVariant +'docs'] =  
df[df['year']==year]['numberOfDocs'].tolist()[0]
```

```
                        neo2015.at[i,str(year)+languageVariant +'subcorpSize'] =  
df[df['year']==year]['subcorpusSize.tokens'].tolist()[0]
```

```
                    else:
```



```
neo2015.at[i,str(year)+languageVariant +'freq'] = 0
neo2015.at[i,str(year)+languageVariant +'docs'] = 0
neo2015.at[i,str(year)+languageVariant +'subcorpSize'] = 0
```

except:

```
print('no response for: '+neo2015.at[i,'lemma'])
```

Wegschrijven

with

```
Path('/media/projecten/Onderwijs/Masterproeven/VanLintElla/neoloogfrequentieverlo
op_volledig.txt').open('w') as f:
```

```
neo2015.to_csv(f, sep='\t', index=False, line_terminator='\n')
```

Appendix IV: Survivaltabellen, chi-squares, p-waarden en pairwise comparisons

Algemeen

```
Call: survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ 1,
              data = wnw)
```

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	3804	1480	0.611	0.00790	0.596	0.627
1.0	2299	180	0.563	0.00805	0.548	0.579
2.0	2085	142	0.525	0.00812	0.509	0.541
3.0	1932	150	0.484	0.00814	0.468	0.500
4.0	1763	146	0.444	0.00812	0.428	0.460
5.0	1605	127	0.409	0.00805	0.393	0.425
6.0	1455	92	0.383	0.00798	0.368	0.399
7.0	1214	64	0.363	0.00795	0.348	0.379
8.0	1063	57	0.343	0.00793	0.328	0.359
9.0	950	48	0.326	0.00791	0.311	0.342
10.0	866	49	0.308	0.00789	0.292	0.323
11.0	760	52	0.286	0.00787	0.271	0.302
12.0	692	72	0.257	0.00780	0.242	0.272
13.0	576	38	0.240	0.00775	0.225	0.255
14.0	495	44	0.218	0.00770	0.204	0.234
15.0	413	48	0.193	0.00763	0.179	0.209
16.0	326	47	0.165	0.00753	0.151	0.181
17.0	242	40	0.138	0.00742	0.124	0.153
18.0	176	1	0.137	0.00742	0.123	0.152

Bron

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ bron, data = wnw)`

bron=neo in neo								
time	n.risk	n.event	survival	std.err	Lower	95% CI	upper	95% CI
0.1	2293	1158	0.4950	0.01044		0.4749		0.5159
1.0	1135	106	0.4488	0.01039		0.4289		0.4696
2.0	1027	93	0.4081	0.01027		0.3885		0.4287
3.0	934	100	0.3644	0.01005		0.3452		0.3847
4.0	829	95	0.3227	0.00977		0.3041		0.3424
5.0	732	89	0.2834	0.00943		0.2655		0.3025
6.0	634	52	0.2602	0.00919		0.2428		0.2788
7.0	461	27	0.2449	0.00911		0.2277		0.2635
8.0	383	21	0.2315	0.00907		0.2144		0.2500
9.0	332	16	0.2204	0.00905		0.2033		0.2388
10.0	300	16	0.2086	0.00903		0.1916		0.2271
11.0	264	24	0.1896	0.00900		0.1728		0.2081
12.0	234	37	0.1597	0.00882		0.1433		0.1779
13.0	190	14	0.1479	0.00872		0.1318		0.1660
14.0	160	14	0.1350	0.00861		0.1191		0.1529
15.0	136	17	0.1181	0.00845		0.1026		0.1359
16.0	106	10	0.1069	0.00836		0.0918		0.1246
17.0	87	20	0.0824	0.00804		0.0680		0.0997

bron=wnw								
time	n.risk	n.event	survival	std.err	Lower	95% CI	upper	95% CI
0.1	1511	322	0.787	0.0105		0.767		0.808
1.0	1164	74	0.737	0.0114		0.715		0.759
2.0	1058	49	0.703	0.0118		0.680		0.726
3.0	998	50	0.668	0.0122		0.644		0.692
4.0	934	51	0.631	0.0126		0.607		0.656
5.0	873	38	0.604	0.0128		0.579		0.629
6.0	821	40	0.574	0.0130		0.549		0.600
7.0	753	37	0.546	0.0132		0.521		0.572
8.0	680	36	0.517	0.0133		0.492		0.544
9.0	618	32	0.490	0.0134		0.465		0.517
10.0	566	33	0.462	0.0136		0.436		0.489
11.0	496	28	0.436	0.0137		0.410		0.463
12.0	458	35	0.402	0.0137		0.376		0.430
13.0	386	24	0.377	0.0138		0.351		0.405
14.0	335	30	0.344	0.0139		0.317		0.372
15.0	277	31	0.305	0.0139		0.279		0.334
16.0	220	37	0.254	0.0139		0.228		0.283
17.0	155	20	0.221	0.0139		0.195		0.250
18.0	119	1	0.219	0.0139		0.194		0.248

Call:
survdiff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ bron,
data = wnw)

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
bron=neo in neo	2293	1909	1483	123	345
bron=wnw	1511	968	1394	130	345

Chisq= 345 on 1 degrees of freedom, p= <2e-16

Lengte

Call: survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ lengte,
data = wnw)

lengte=kort							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	451	103	0.772	0.0198		0.734	0.811
1.0	342	20	0.726	0.0210		0.686	0.769
2.0	314	12	0.699	0.0217		0.657	0.743
3.0	302	21	0.650	0.0226		0.607	0.696
4.0	278	26	0.589	0.0234		0.545	0.637
5.0	249	10	0.566	0.0237		0.521	0.614
6.0	234	13	0.534	0.0239		0.489	0.583
7.0	203	11	0.505	0.0242		0.460	0.555
8.0	180	5	0.491	0.0243		0.446	0.541
9.0	165	9	0.464	0.0245		0.419	0.515
10.0	150	3	0.455	0.0246		0.409	0.506
11.0	133	6	0.435	0.0249		0.388	0.486
12.0	125	5	0.417	0.0251		0.371	0.469
13.0	108	6	0.394	0.0254		0.347	0.447
14.0	91	7	0.364	0.0259		0.316	0.418
15.0	76	6	0.335	0.0264		0.287	0.391
16.0	64	9	0.288	0.0270		0.240	0.346
17.0	49	5	0.259	0.0272		0.210	0.318

lengte=lang							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	1373	663	0.5171	0.0135		0.4913	0.544
1.0	707	60	0.4732	0.0135		0.4475	0.500
2.0	636	46	0.4390	0.0134		0.4135	0.466
3.0	587	53	0.3994	0.0133		0.3742	0.426
4.0	528	51	0.3608	0.0130		0.3361	0.387
5.0	477	49	0.3237	0.0127		0.2997	0.350
6.0	421	30	0.3007	0.0125		0.2771	0.326
7.0	336	24	0.2792	0.0123		0.2560	0.304
8.0	290	15	0.2647	0.0123		0.2418	0.290
9.0	262	14	0.2506	0.0122		0.2278	0.276
10.0	239	16	0.2338	0.0121		0.2113	0.259
11.0	215	10	0.2229	0.0120		0.2007	0.248
12.0	199	28	0.1916	0.0117		0.1700	0.216
13.0	160	8	0.1820	0.0116		0.1607	0.206
14.0	142	14	0.1641	0.0114		0.1432	0.188
15.0	115	16	0.1412	0.0111		0.1210	0.165
16.0	88	14	0.1188	0.0109		0.0993	0.142
17.0	71	12	0.0987	0.0105		0.0802	0.121
18.0	52	1	0.0968	0.0104		0.0784	0.120

lengte=medium							
time	n.risk	n.event	survival	std.err	lower	95% CI	upper 95% CI
0.1	1980	714	0.639	0.0108		0.619	0.661
1.0	1250	100	0.588	0.0111		0.567	0.610
2.0	1135	84	0.545	0.0112		0.523	0.567
3.0	1043	76	0.505	0.0113		0.483	0.528
4.0	957	69	0.469	0.0113		0.447	0.491
5.0	879	68	0.432	0.0112		0.411	0.455
6.0	800	49	0.406	0.0112		0.385	0.428
7.0	675	29	0.388	0.0112		0.367	0.411
8.0	593	37	0.364	0.0111		0.343	0.387
9.0	523	25	0.347	0.0111		0.326	0.369
10.0	477	30	0.325	0.0111		0.304	0.348
11.0	412	36	0.297	0.0111		0.276	0.319
12.0	368	39	0.265	0.0110		0.244	0.288
13.0	308	24	0.244	0.0109		0.224	0.267
14.0	262	23	0.223	0.0109		0.203	0.245
15.0	222	26	0.197	0.0107		0.177	0.219
16.0	174	24	0.170	0.0106		0.150	0.192
17.0	122	23	0.138	0.0105		0.119	0.160

Pairwise comparisons using Log-Rank test

data: wnw and lengte

	kort	lang
lang	< 2e-16	-
medium	1.6e-09	1.2e-09

P value adjustment method: BH

Call:

```
survdifff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ lengte,
           data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
lengte=kort	451	277	406	40.76	60.75
lengte=lang	1373	1124	944	34.40	68.15
lengte=medium	1980	1476	1528	1.74	4.88

Chisq= 101 on 2 degrees of freedom, p= <2e-16

Levenshteinafstand

Call: survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ afstand,
data = wnw)

afstand=hoog								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	2658	1146	0.569	0.00961		0.550		0.588
1.0	1497	125	0.521	0.00970		0.503		0.541
2.0	1351	94	0.485	0.00972		0.466		0.505
3.0	1248	102	0.445	0.00968		0.427		0.465
4.0	1133	91	0.410	0.00961		0.391		0.429
5.0	1036	92	0.373	0.00947		0.355		0.392
6.0	932	63	0.348	0.00935		0.330		0.367
7.0	761	41	0.329	0.00929		0.312		0.348
8.0	664	41	0.309	0.00925		0.291		0.328
9.0	589	30	0.293	0.00921		0.276		0.312
10.0	541	35	0.274	0.00916		0.257		0.293
11.0	475	35	0.254	0.00910		0.237		0.273
12.0	430	53	0.223	0.00893		0.206		0.241
13.0	352	22	0.209	0.00886		0.192		0.227
14.0	307	29	0.189	0.00874		0.173		0.207
15.0	256	30	0.167	0.00860		0.151		0.185
16.0	201	28	0.144	0.00845		0.128		0.161
17.0	155	29	0.117	0.00821		0.102		0.134
18.0	111	1	0.116	0.00821		0.101		0.133

afstand=laag								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	164	34	0.793	0.0317		0.733		0.857
1.0	130	5	0.762	0.0332		0.700		0.830
2.0	121	7	0.718	0.0353		0.652		0.791
3.0	113	7	0.674	0.0369		0.605		0.750
4.0	106	10	0.610	0.0385		0.539		0.690
5.0	96	4	0.585	0.0389		0.513		0.666
6.0	90	4	0.559	0.0393		0.487		0.641
7.0	81	6	0.517	0.0398		0.445		0.602
8.0	73	3	0.496	0.0401		0.423		0.581
9.0	67	3	0.474	0.0403		0.401		0.560
10.0	61	1	0.466	0.0403		0.393		0.552
11.0	54	2	0.449	0.0407		0.376		0.536
12.0	49	2	0.430	0.0410		0.357		0.519
13.0	41	2	0.409	0.0416		0.336		0.500
14.0	36	2	0.387	0.0423		0.312		0.479
15.0	31	2	0.362	0.0431		0.286		0.457
16.0	28	4	0.310	0.0440		0.235		0.410
17.0	21	2	0.281	0.0445		0.206		0.383

afstand=medium						
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	982	300	0.695	0.0147	0.666	0.724
1.0	672	50	0.643	0.0153	0.614	0.674
2.0	613	41	0.600	0.0157	0.570	0.631
3.0	571	41	0.557	0.0159	0.526	0.589
4.0	524	45	0.509	0.0161	0.478	0.541
5.0	473	31	0.476	0.0161	0.445	0.508
6.0	433	25	0.448	0.0161	0.418	0.481
7.0	372	17	0.428	0.0161	0.397	0.460
8.0	326	13	0.411	0.0161	0.380	0.443
9.0	294	15	0.390	0.0162	0.359	0.423
10.0	264	13	0.370	0.0163	0.340	0.404
11.0	231	15	0.346	0.0163	0.316	0.380
12.0	213	17	0.319	0.0164	0.288	0.352
13.0	183	14	0.294	0.0163	0.264	0.328
14.0	152	13	0.269	0.0164	0.239	0.303
15.0	126	16	0.235	0.0164	0.205	0.269
16.0	97	15	0.199	0.0163	0.169	0.233
17.0	66	9	0.172	0.0164	0.142	0.207

Pairwise comparisons using Log-Rank test

data: wnw and afstand

	hoog	laag
laag	1.9e-08	-
medium	9.3e-09	0.0025

P value adjustment method: BH

Call:

```
survdifff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ afstand,
           data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
afstand=hoog	2658	2087	1925	13.7	53.9
afstand=laag	164	100	154	18.9	25.2
afstand=medium	982	690	798	14.7	26.5

Chisq= 61.2 on 2 degrees of freedom, p= 5e-14

Gewogen Levenshteinafstand

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ gewogenLevenshtein, data = wnw)`

gewogenLevenshtein=hoog							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	1825	781	0.572	0.0116		0.5498	0.595
1.0	1033	68	0.534	0.0117		0.5120	0.558
2.0	947	70	0.495	0.0117		0.4724	0.518
3.0	870	68	0.456	0.0117		0.4338	0.480
4.0	792	65	0.419	0.0116		0.3966	0.442
5.0	720	67	0.380	0.0115		0.3579	0.403
6.0	643	45	0.353	0.0113		0.3317	0.376
7.0	521	29	0.334	0.0113		0.3122	0.356
8.0	453	21	0.318	0.0113		0.2968	0.341
9.0	409	24	0.299	0.0112		0.2782	0.322
10.0	372	21	0.283	0.0112		0.2614	0.305
11.0	327	22	0.264	0.0111		0.2426	0.286
12.0	297	37	0.231	0.0110		0.2101	0.253
13.0	242	16	0.215	0.0109		0.1951	0.238
14.0	210	18	0.197	0.0108		0.1769	0.219
15.0	175	23	0.171	0.0106		0.1514	0.193
16.0	134	15	0.152	0.0105		0.1326	0.174
17.0	104	23	0.118	0.0103		0.0998	0.140

gewogenLevenshtein=laag							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	193	54	0.720	0.0323		0.660	0.786
1.0	139	9	0.674	0.0338		0.611	0.743
2.0	126	9	0.625	0.0349		0.561	0.698
3.0	116	8	0.582	0.0357		0.516	0.657
4.0	108	12	0.518	0.0363		0.451	0.594
5.0	96	7	0.480	0.0363		0.414	0.557
6.0	88	7	0.442	0.0362		0.376	0.519
7.0	76	5	0.413	0.0361		0.348	0.490
8.0	66	3	0.394	0.0360		0.329	0.471
9.0	59	2	0.381	0.0360		0.316	0.458
10.0	54	1	0.373	0.0360		0.309	0.451
11.0	50	2	0.359	0.0361		0.294	0.437
12.0	46	2	0.343	0.0362		0.279	0.422
13.0	38	2	0.325	0.0365		0.261	0.405
14.0	34	3	0.296	0.0368		0.232	0.378
15.0	29	2	0.276	0.0370		0.212	0.359
16.0	24	3	0.241	0.0373		0.178	0.327
17.0	18	2	0.215	0.0377		0.152	0.303

gewogenLevenshtein=medium

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	1786	645	0.639	0.0114	0.617	0.662
1.0	1127	103	0.580	0.0117	0.558	0.604
2.0	1012	63	0.544	0.0118	0.522	0.568
3.0	946	74	0.502	0.0119	0.479	0.526
4.0	863	69	0.462	0.0119	0.439	0.486
5.0	789	53	0.431	0.0118	0.408	0.454
6.0	724	40	0.407	0.0117	0.384	0.431
7.0	617	30	0.387	0.0117	0.365	0.411
8.0	544	33	0.364	0.0117	0.341	0.387
9.0	482	22	0.347	0.0117	0.325	0.371
10.0	440	27	0.326	0.0117	0.304	0.349
11.0	383	28	0.302	0.0116	0.280	0.326
12.0	349	33	0.273	0.0116	0.252	0.297
13.0	296	20	0.255	0.0115	0.233	0.278
14.0	251	23	0.232	0.0114	0.210	0.255
15.0	209	23	0.206	0.0113	0.185	0.229
16.0	168	29	0.170	0.0111	0.150	0.194
17.0	120	15	0.149	0.0110	0.129	0.172
18.0	90	1	0.148	0.0110	0.127	0.171

Pairwise comparisons using Log-Rank test

data: wnw and gewogenLevenshtein

	hoog	laag
laag	0.0019	-
medium	0.0019	0.0464

P value adjustment method: BH

Call:

```
survdifff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ gewogenLevenshtein,
  data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
gewogenLevenshtein=hoog	1825	1413	1325	5.86	14.27
gewogenLevenshtein=laag	193	133	163	5.43	7.39
gewogenLevenshtein=medium	1786	1331	1389	2.46	6.22

Chisq= 17.9 on 2 degrees of freedom, p= 1e-04

Frequentie dichtstbijzijnde woord

Call: survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ levfreq,
data = wnw)

levfreq=hoog

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
4	12	1	0.917	0.0798	0.773	1
8	10	1	0.825	0.1128	0.631	1
12	5	1	0.660	0.1730	0.395	1

levfreq=laag

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	122	38	0.689	0.0419	0.611	0.776
1.0	84	6	0.639	0.0435	0.560	0.730
2.0	74	8	0.570	0.0451	0.488	0.666
3.0	65	3	0.544	0.0455	0.462	0.641
4.0	62	9	0.465	0.0459	0.383	0.564
5.0	53	7	0.404	0.0453	0.324	0.503
6.0	46	5	0.360	0.0444	0.282	0.458
7.0	38	3	0.331	0.0439	0.256	0.429
9.0	32	1	0.321	0.0437	0.246	0.419
10.0	29	1	0.310	0.0436	0.235	0.408
11.0	25	2	0.285	0.0435	0.211	0.384
14.0	21	3	0.244	0.0431	0.173	0.345
15.0	17	1	0.230	0.0429	0.160	0.332
16.0	13	1	0.212	0.0431	0.143	0.316
17.0	10	2	0.170	0.0437	0.103	0.281

levfreq=medium

time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	59	16	0.729	0.0579	0.624	0.852
1.0	43	3	0.678	0.0608	0.569	0.808
2.0	40	1	0.661	0.0616	0.551	0.794
3.0	39	5	0.576	0.0643	0.463	0.717
4.0	34	2	0.542	0.0649	0.429	0.686
6.0	31	2	0.507	0.0652	0.394	0.653
7.0	28	2	0.471	0.0654	0.359	0.618
8.0	24	2	0.432	0.0656	0.321	0.582
9.0	21	1	0.411	0.0656	0.301	0.562
12.0	18	1	0.388	0.0658	0.279	0.541
13.0	14	2	0.333	0.0671	0.224	0.494
15.0	10	1	0.300	0.0682	0.192	0.468
16.0	9	2	0.233	0.0673	0.132	0.411

levfreq=na								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	3611	1426	0.605	0.00813		0.589		0.621
1.0	2160	171	0.557	0.00827		0.541		0.574
2.0	1959	133	0.519	0.00834		0.503		0.536
3.0	1816	142	0.479	0.00835		0.463		0.495
4.0	1655	134	0.440	0.00832		0.424		0.457
5.0	1509	120	0.405	0.00825		0.389		0.421
6.0	1367	85	0.380	0.00818		0.364		0.396
7.0	1138	59	0.360	0.00814		0.345		0.376
8.0	997	54	0.341	0.00812		0.325		0.357
9.0	891	46	0.323	0.00811		0.308		0.339
10.0	812	48	0.304	0.00808		0.289		0.320
11.0	710	50	0.283	0.00806		0.267		0.299
12.0	646	70	0.252	0.00798		0.237		0.268
13.0	538	36	0.235	0.00792		0.220		0.251
14.0	461	41	0.214	0.00786		0.199		0.230
15.0	384	46	0.189	0.00778		0.174		0.204
16.0	302	44	0.161	0.00767		0.147		0.177
17.0	224	38	0.134	0.00754		0.120		0.149
18.0	161	1	0.133	0.00754		0.119		0.149

Pairwise comparisons using Log-Rank test

data: wnw and levfreq

	hoog	laag	medium
laag	0.0082	-	-
medium	0.0444	0.2499	-
na	0.0080	0.4055	0.0590

P value adjustment method: BH

Call:

```
survdif(formula = Surv(leeftijd, leeft.nog.in.2021) ~ levfreq,  
data = wnw)
```

	N	Observed	Expected	(O-E) ² /E	(O-E) ² /V
levfreq=hoog	12	3	13.6	8.266	10.162
levfreq=laag	122	90	96.2	0.405	0.545
levfreq=medium	59	40	52.9	3.139	4.057
levfreq=na	3611	2744	2714.3	0.326	7.395

Chisq= 15.1 on 3 degrees of freedom, p= 0.002

Lidwoord

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Lidwoord, data = wnw)`

Lidwoord=de							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	1041	220	0.789	0.0127	0.764	0.814	
1.0	803	52	0.738	0.0137	0.711	0.765	
2.0	730	34	0.703	0.0142	0.676	0.732	
3.0	688	33	0.670	0.0147	0.641	0.699	
4.0	648	41	0.627	0.0152	0.598	0.658	
5.0	599	29	0.597	0.0155	0.567	0.628	
6.0	562	30	0.565	0.0157	0.535	0.597	
7.0	513	23	0.540	0.0159	0.509	0.572	
8.0	462	24	0.512	0.0160	0.481	0.544	
9.0	419	23	0.483	0.0162	0.453	0.516	
10.0	383	24	0.453	0.0163	0.422	0.486	
11.0	336	16	0.432	0.0164	0.401	0.465	
12.0	313	26	0.396	0.0165	0.365	0.429	
13.0	264	13	0.376	0.0165	0.345	0.410	
14.0	232	23	0.339	0.0166	0.308	0.373	
15.0	189	22	0.300	0.0167	0.269	0.334	
16.0	151	28	0.244	0.0166	0.214	0.279	
17.0	105	12	0.216	0.0165	0.186	0.251	
18.0	83	1	0.213	0.0165	0.183	0.248	

Lidwoord=de of het							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	43	5	0.884	0.0489	0.793	0.985	
1.0	37	1	0.860	0.0531	0.762	0.970	
2.0	35	1	0.835	0.0570	0.731	0.955	
3.0	34	1	0.811	0.0604	0.701	0.938	
4.0	33	1	0.786	0.0633	0.671	0.921	
6.0	31	2	0.735	0.0687	0.612	0.883	
7.0	28	3	0.657	0.0749	0.525	0.821	
8.0	25	1	0.630	0.0763	0.497	0.799	
10.0	24	1	0.604	0.0775	0.470	0.777	
11.0	22	1	0.577	0.0787	0.441	0.754	
12.0	21	2	0.522	0.0802	0.386	0.705	
13.0	18	2	0.464	0.0811	0.329	0.653	
14.0	15	1	0.433	0.0814	0.299	0.626	
15.0	11	1	0.393	0.0830	0.260	0.595	
16.0	8	1	0.344	0.0859	0.211	0.562	

Lidwoord=geen							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	178	35	0.803	0.0298		0.747	0.864
1.0	139	6	0.769	0.0317		0.709	0.833
2.0	127	3	0.751	0.0326		0.689	0.817
3.0	121	7	0.707	0.0346		0.642	0.778
4.0	109	5	0.675	0.0359		0.608	0.749
5.0	103	3	0.655	0.0366		0.587	0.731
6.0	100	6	0.616	0.0378		0.546	0.694
7.0	91	7	0.568	0.0389		0.497	0.650
8.0	80	3	0.547	0.0393		0.475	0.630
9.0	74	5	0.510	0.0400		0.437	0.595
10.0	65	2	0.494	0.0403		0.421	0.580
11.0	55	4	0.458	0.0412		0.384	0.547
12.0	49	3	0.430	0.0417		0.356	0.520
13.0	41	4	0.388	0.0426		0.313	0.482
14.0	34	2	0.366	0.0431		0.290	0.460
15.0	28	4	0.313	0.0441		0.238	0.413
16.0	19	2	0.280	0.0452		0.204	0.385
17.0	15	2	0.243	0.0463		0.167	0.353

Lidwoord=het							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	249	62	0.751	0.0274		0.699	0.807
1.0	185	15	0.690	0.0293		0.635	0.750
2.0	166	11	0.644	0.0305		0.587	0.707
3.0	155	9	0.607	0.0311		0.549	0.671
4.0	144	4	0.590	0.0314		0.532	0.655
5.0	139	6	0.565	0.0317		0.506	0.630
6.0	128	2	0.556	0.0318		0.497	0.622
7.0	121	4	0.537	0.0321		0.478	0.604
8.0	113	8	0.499	0.0325		0.440	0.567
9.0	101	4	0.480	0.0327		0.420	0.548
10.0	94	6	0.449	0.0329		0.389	0.518
11.0	83	7	0.411	0.0331		0.351	0.481
12.0	75	4	0.389	0.0331		0.329	0.460
13.0	63	5	0.358	0.0332		0.299	0.430
14.0	54	4	0.332	0.0333		0.273	0.404
15.0	49	4	0.305	0.0332		0.246	0.377
16.0	42	6	0.261	0.0329		0.204	0.334
17.0	30	6	0.209	0.0325		0.154	0.283

Lidwoord=na								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	2293	1158	0.4950	0.01044		0.4749		0.5159
1.0	1135	106	0.4488	0.01039		0.4289		0.4696
2.0	1027	93	0.4081	0.01027		0.3885		0.4287
3.0	934	100	0.3644	0.01005		0.3452		0.3847
4.0	829	95	0.3227	0.00977		0.3041		0.3424
5.0	732	89	0.2834	0.00943		0.2655		0.3025
6.0	634	52	0.2602	0.00919		0.2428		0.2788
7.0	461	27	0.2449	0.00911		0.2277		0.2635
8.0	383	21	0.2315	0.00907		0.2144		0.2500
9.0	332	16	0.2204	0.00905		0.2033		0.2388
10.0	300	16	0.2086	0.00903		0.1916		0.2271
11.0	264	24	0.1896	0.00900		0.1728		0.2081
12.0	234	37	0.1597	0.00882		0.1433		0.1779
13.0	190	14	0.1479	0.00872		0.1318		0.1660
14.0	160	14	0.1350	0.00861		0.1191		0.1529
15.0	136	17	0.1181	0.00845		0.1026		0.1359
16.0	106	10	0.1069	0.00836		0.0918		0.1246
17.0	87	20	0.0824	0.00804		0.0680		0.0997

Pairwise comparisons using Log-Rank test

data: wnw and Lidwoord

	de	de of het	geen	het
de of het	0.17	-	-	-
geen	0.37	0.35	-	-
het	0.65	0.17	0.35	-
na	< 2e-16	9.7e-07	1.0e-15	1.6e-15

P value adjustment method: BH

Call:
 survdiff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ Lidwoord,
 data = wnw)

	N	Observed	Expected	(O-E) ² /E	(O-E) ² /V
Lidwoord=de	1041	674	958.3	84.3	166.0
Lidwoord=de of het	43	24	45.7	10.3	13.0
Lidwoord=geen	178	103	162.2	21.6	29.0
Lidwoord=het	249	167	228.2	16.4	22.7
Lidwoord=na	2293	1909	1482.7	122.6	345.1

Chisq= 348 on 4 degrees of freedom, p= <2e-16

Geslacht

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Geslacht2, data = wnw)`

Geslacht2=mannelijk								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	537	123	0.771	0.0181		0.736		0.807
1.0	403	25	0.723	0.0194		0.686		0.762
2.0	371	13	0.698	0.0199		0.660		0.738
3.0	356	16	0.666	0.0205		0.627		0.708
4.0	335	21	0.625	0.0212		0.585		0.668
5.0	311	16	0.593	0.0215		0.552		0.636
6.0	293	15	0.562	0.0218		0.521		0.607
7.0	268	14	0.533	0.0220		0.491		0.578
8.0	239	12	0.506	0.0222		0.464		0.552
9.0	217	13	0.476	0.0224		0.434		0.522
10.0	195	10	0.451	0.0226		0.409		0.498
11.0	173	7	0.433	0.0227		0.391		0.480
12.0	164	13	0.399	0.0228		0.356		0.446
13.0	137	7	0.378	0.0229		0.336		0.426
14.0	116	9	0.349	0.0231		0.307		0.397
15.0	93	9	0.315	0.0235		0.272		0.365
16.0	75	14	0.256	0.0238		0.214		0.308
17.0	52	6	0.227	0.0239		0.184		0.279

Geslacht2=na								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	2293	1158	0.4950	0.01044		0.4749		0.5159
1.0	1135	106	0.4488	0.01039		0.4289		0.4696
2.0	1027	93	0.4081	0.01027		0.3885		0.4287
3.0	934	100	0.3644	0.01005		0.3452		0.3847
4.0	829	95	0.3227	0.00977		0.3041		0.3424
5.0	732	89	0.2834	0.00943		0.2655		0.3025
6.0	634	52	0.2602	0.00919		0.2428		0.2788
7.0	461	27	0.2449	0.00911		0.2277		0.2635
8.0	383	21	0.2315	0.00907		0.2144		0.2500
9.0	332	16	0.2204	0.00905		0.2033		0.2388
10.0	300	16	0.2086	0.00903		0.1916		0.2271
11.0	264	24	0.1896	0.00900		0.1728		0.2081
12.0	234	37	0.1597	0.00882		0.1433		0.1779
13.0	190	14	0.1479	0.00872		0.1318		0.1660
14.0	160	14	0.1350	0.00861		0.1191		0.1529
15.0	136	17	0.1181	0.00845		0.1026		0.1359
16.0	106	10	0.1069	0.00836		0.0918		0.1246
17.0	87	20	0.0824	0.00804		0.0680		0.0997

Geslacht2=nvt							
time	n.risk	n.event	survival	std.err	lower	95% CI	upper 95% CI
0.1	172	34	0.802	0.0304		0.745	0.864
1.0	135	6	0.767	0.0323		0.706	0.833
2.0	123	2	0.754	0.0330		0.692	0.822
3.0	118	7	0.709	0.0351		0.644	0.782
4.0	108	5	0.677	0.0364		0.609	0.752
5.0	102	3	0.657	0.0371		0.588	0.734
6.0	99	6	0.617	0.0382		0.546	0.697
7.0	90	7	0.569	0.0393		0.497	0.652
8.0	79	3	0.547	0.0398		0.475	0.631
9.0	73	4	0.517	0.0403		0.444	0.603
10.0	65	2	0.501	0.0406		0.428	0.588
11.0	54	4	0.464	0.0416		0.389	0.554
12.0	48	3	0.435	0.0423		0.360	0.527
13.0	40	4	0.392	0.0433		0.315	0.486
14.0	33	2	0.368	0.0438		0.291	0.465
15.0	27	3	0.327	0.0448		0.250	0.428
16.0	19	2	0.293	0.0463		0.215	0.399
17.0	15	2	0.254	0.0476		0.176	0.366

Geslacht2=onzeker							
time	n.risk	n.event	survival	std.err	lower	95% CI	upper 95% CI
0.1	304	54	0.822	0.0219		0.781	0.866
1.0	244	10	0.789	0.0235		0.744	0.836
2.0	228	14	0.740	0.0253		0.692	0.792
3.0	210	13	0.694	0.0268		0.644	0.749
4.0	195	12	0.652	0.0278		0.599	0.709
5.0	178	6	0.630	0.0283		0.577	0.688
6.0	167	8	0.600	0.0289		0.546	0.659
7.0	151	7	0.572	0.0294		0.517	0.632
8.0	136	7	0.542	0.0299		0.487	0.604
9.0	123	1	0.538	0.0300		0.482	0.600
10.0	118	10	0.492	0.0307		0.436	0.556
11.0	104	7	0.459	0.0311		0.402	0.524
12.0	94	7	0.425	0.0314		0.368	0.491
13.0	80	6	0.393	0.0316		0.336	0.460
14.0	70	8	0.348	0.0317		0.291	0.416
15.0	57	7	0.305	0.0317		0.249	0.374
16.0	45	6	0.265	0.0315		0.210	0.334
17.0	30	4	0.229	0.0319		0.175	0.301

Geslacht2=onzijdig							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	254	62	0.756	0.0270	0.705	0.811	
1.0	189	16	0.692	0.0290	0.637	0.751	
2.0	169	11	0.647	0.0302	0.590	0.709	
3.0	158	9	0.610	0.0308	0.552	0.674	
4.0	146	4	0.593	0.0311	0.535	0.658	
5.0	141	6	0.568	0.0314	0.510	0.633	
6.0	129	2	0.559	0.0316	0.501	0.625	
7.0	122	4	0.541	0.0318	0.482	0.607	
8.0	114	8	0.503	0.0323	0.443	0.570	
9.0	102	5	0.478	0.0325	0.419	0.547	
10.0	94	6	0.448	0.0328	0.388	0.517	
11.0	83	7	0.410	0.0330	0.350	0.480	
12.0	75	4	0.388	0.0330	0.329	0.458	
13.0	63	5	0.357	0.0331	0.298	0.429	
14.0	54	4	0.331	0.0332	0.272	0.403	
15.0	49	4	0.304	0.0331	0.245	0.376	
16.0	42	6	0.260	0.0328	0.204	0.333	
17.0	30	6	0.208	0.0324	0.154	0.283	

Geslacht2=vrouwelijk							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	244	49	0.799	0.0256	0.750	0.851	
1.0	193	17	0.729	0.0285	0.675	0.787	
2.0	167	9	0.690	0.0298	0.633	0.751	
3.0	156	5	0.667	0.0305	0.610	0.730	
4.0	150	9	0.627	0.0314	0.569	0.692	
5.0	141	7	0.596	0.0320	0.537	0.662	
6.0	133	9	0.556	0.0325	0.496	0.623	
7.0	122	5	0.533	0.0328	0.473	0.601	
8.0	112	6	0.505	0.0330	0.444	0.574	
9.0	103	9	0.460	0.0332	0.400	0.530	
10.0	94	5	0.436	0.0332	0.375	0.506	
11.0	82	3	0.420	0.0333	0.360	0.491	
12.0	77	8	0.376	0.0332	0.317	0.447	
13.0	66	2	0.365	0.0332	0.305	0.436	
14.0	62	7	0.324	0.0329	0.265	0.395	
15.0	51	8	0.273	0.0322	0.217	0.344	
16.0	39	9	0.210	0.0309	0.157	0.280	
17.0	28	2	0.195	0.0305	0.144	0.265	
18.0	25	1	0.187	0.0302	0.136	0.257	

Pairwise comparisons using Log-Rank test

data: wnw and Geslacht2

	mannelijk	na	nvt	onzeker	onzijdig
na	< 2e-16	-	-	-	-
nvt	0.47	8.7e-16	-	-	-
onzeker	0.59	< 2e-16	0.76	-	-
onzijdig	0.76	8.7e-16	0.47	0.47	-
vrouwelijk	0.74	3.7e-15	0.47	0.47	0.91

P value adjustment method: BH

Call:

```
survdif(formula = Surv(leeftijd, leeft.nog.in.2021) ~ Geslacht2,
  data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
Geslacht2=mannelijk	537	343	491	44.4	68.6
Geslacht2=na	2293	1909	1483	122.6	345.1
Geslacht2=nvt	172	99	158	22.1	29.6
Geslacht2=onzeker	304	187	285	33.7	47.4
Geslacht2=onzijdig	254	169	231	16.8	23.3
Geslacht2=vrouwelijk	244	170	229	15.4	21.2

Chisq= 348 on 5 degrees of freedom, p= <2e-16

Woordsoort

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Woordsoort, data = wnw)`

Woordsoort=adjectief							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	74	26	0.649	0.0555		0.549	0.767
1.0	46	3	0.606	0.0570		0.504	0.729
2.0	43	2	0.578	0.0577		0.475	0.703
3.0	40	6	0.491	0.0589		0.388	0.622
4.0	34	3	0.448	0.0588		0.346	0.580
5.0	30	2	0.418	0.0586		0.318	0.550
6.0	28	2	0.388	0.0581		0.290	0.521
8.0	18	1	0.367	0.0587		0.268	0.502
11.0	16	3	0.298	0.0596		0.201	0.441
13.0	13	1	0.275	0.0593		0.180	0.420
17.0	9	1	0.244	0.0601		0.151	0.396

Woordsoort=overige							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	32	9	0.719	0.0795		0.5787	0.893
1.0	23	2	0.656	0.0840		0.5107	0.843
2.0	19	1	0.622	0.0864		0.4735	0.816
3.0	18	2	0.553	0.0895		0.4023	0.759
4.0	15	1	0.516	0.0908		0.3653	0.728
6.0	14	1	0.479	0.0915		0.3294	0.696
9.0	11	1	0.435	0.0930		0.2865	0.662
10.0	10	1	0.392	0.0933		0.2457	0.625
12.0	8	1	0.343	0.0936		0.2008	0.586
17.0	4	2	0.171	0.0977		0.0561	0.524

Woordsoort=substantief

time	n.risk	n.event	survival	std.err	Lower 95% CI	upper 95% CI
0.1	3560	1415	0.603	0.00820	0.587	0.619
1.0	2122	170	0.554	0.00834	0.538	0.571
2.0	1924	137	0.515	0.00840	0.499	0.532
3.0	1779	137	0.475	0.00841	0.459	0.492
4.0	1625	136	0.435	0.00837	0.419	0.452
5.0	1479	121	0.400	0.00829	0.384	0.416
6.0	1335	84	0.375	0.00821	0.359	0.391
7.0	1111	57	0.355	0.00817	0.340	0.372
8.0	973	53	0.336	0.00815	0.320	0.352
9.0	868	44	0.319	0.00813	0.303	0.335
10.0	791	47	0.300	0.00810	0.285	0.316
11.0	695	46	0.280	0.00808	0.265	0.296
12.0	635	70	0.249	0.00799	0.234	0.265
13.0	527	34	0.233	0.00793	0.218	0.249
14.0	452	42	0.212	0.00787	0.197	0.228
15.0	376	45	0.186	0.00778	0.172	0.202
16.0	296	45	0.158	0.00766	0.144	0.174
17.0	218	36	0.132	0.00753	0.118	0.147
18.0	158	1	0.131	0.00752	0.117	0.147

Woordsoort=werkwoord

time	n.risk	n.event	survival	std.err	Lower 95% CI	upper 95% CI
0.1	138	30	0.783	0.0351	0.717	0.855
1.0	108	5	0.746	0.0370	0.677	0.823
2.0	99	2	0.731	0.0378	0.661	0.809
3.0	95	5	0.693	0.0395	0.620	0.775
4.0	89	6	0.646	0.0412	0.570	0.732
5.0	82	4	0.615	0.0421	0.537	0.703
6.0	78	5	0.575	0.0429	0.497	0.666
7.0	70	7	0.518	0.0438	0.439	0.611
8.0	61	3	0.492	0.0440	0.413	0.587
9.0	54	3	0.465	0.0443	0.386	0.560
10.0	49	1	0.455	0.0444	0.376	0.551
11.0	41	3	0.422	0.0452	0.342	0.521
12.0	36	1	0.410	0.0454	0.330	0.510
13.0	29	3	0.368	0.0469	0.287	0.472
14.0	25	2	0.338	0.0475	0.257	0.446
15.0	20	3	0.288	0.0486	0.207	0.401
16.0	15	2	0.249	0.0491	0.169	0.367
17.0	11	1	0.227	0.0496	0.148	0.348

Pairwise comparisons using Log-Rank test

data: wnw and Woordsoort

	adjectief	overige	substantief
overige	0.68316	-	-
substantief	0.26400	0.26400	-
werkwoord	0.26400	0.68316	0.00028

P value adjustment method: BH

Call:

```
survdiff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ Woordsoort,  
data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
Woordsoort=adjectief	74	50	58.1	1.12	1.49
Woordsoort=overige	32	21	27.0	1.34	1.74
Woordsoort=substantief	3560	2720	2667.0	1.06	18.51
Woordsoort=werkwoord	138	86	125.0	12.15	16.10

Chisq= 19.9 on 3 degrees of freedom, p= 2e-04

Woordvorming/Samenstelling

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Samenstelling, data = wnw)`

Samenstelling=geen samenstelling						
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	432	114	0.736	0.0212	0.696	0.779
1.0	313	12	0.708	0.0219	0.666	0.752
2.0	290	16	0.669	0.0228	0.626	0.715
3.0	273	16	0.630	0.0234	0.585	0.677
4.0	251	17	0.587	0.0240	0.542	0.636
5.0	230	9	0.564	0.0243	0.518	0.614
6.0	217	13	0.530	0.0246	0.484	0.581
7.0	186	8	0.507	0.0248	0.461	0.558
8.0	166	4	0.495	0.0249	0.449	0.547
9.0	157	10	0.464	0.0253	0.417	0.516
10.0	140	4	0.450	0.0254	0.403	0.503
11.0	125	6	0.429	0.0257	0.381	0.482
12.0	116	6	0.407	0.0259	0.359	0.461
13.0	96	5	0.385	0.0262	0.337	0.440
14.0	83	1	0.381	0.0263	0.333	0.436
15.0	76	3	0.366	0.0267	0.317	0.422
16.0	67	6	0.333	0.0274	0.283	0.391
17.0	51	8	0.281	0.0287	0.230	0.343

Samenstelling=samenstelling						
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	3372	1366	0.595	0.00845	0.579	0.612
1.0	1986	168	0.545	0.00858	0.528	0.562
2.0	1795	126	0.506	0.00863	0.490	0.524
3.0	1659	134	0.465	0.00863	0.449	0.483
4.0	1512	129	0.426	0.00857	0.409	0.443
5.0	1375	118	0.389	0.00847	0.373	0.406
6.0	1238	79	0.364	0.00838	0.348	0.381
7.0	1028	56	0.345	0.00833	0.329	0.361
8.0	897	53	0.324	0.00829	0.308	0.341
9.0	793	38	0.309	0.00827	0.293	0.325
10.0	726	45	0.289	0.00823	0.274	0.306
11.0	635	46	0.269	0.00820	0.253	0.285
12.0	576	66	0.238	0.00809	0.222	0.254
13.0	480	33	0.221	0.00802	0.206	0.238
14.0	412	43	0.198	0.00792	0.183	0.214
15.0	337	45	0.172	0.00778	0.157	0.188
16.0	259	41	0.145	0.00762	0.130	0.160
17.0	191	32	0.120	0.00745	0.107	0.136
18.0	138	1	0.120	0.00745	0.106	0.135

Call:

```
survdiff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ Samenstelling,  
data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
Samenstelling=geen samenstelling	432	258	381	39.71	58.7
Samenstelling=samenstelling	3372	2619	2496	6.06	58.7

Chisq= 58.7 on 1 degrees of freedom, p= 2e-14

Productiviteit

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ productiviteit, data = wnw)`

productiviteit=hoog							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	352	63	0.821	0.0204	0.782	0.862	
1.0	282	20	0.763	0.0228	0.719	0.809	
2.0	256	15	0.718	0.0242	0.672	0.767	
3.0	236	9	0.691	0.0249	0.644	0.741	
4.0	225	10	0.660	0.0256	0.612	0.712	
5.0	213	10	0.629	0.0262	0.580	0.683	
6.0	201	7	0.607	0.0266	0.557	0.662	
7.0	187	4	0.594	0.0268	0.544	0.649	
8.0	176	5	0.577	0.0271	0.527	0.633	
9.0	165	7	0.553	0.0275	0.501	0.609	
10.0	153	10	0.517	0.0280	0.465	0.574	
11.0	136	8	0.486	0.0283	0.434	0.545	
12.0	126	11	0.444	0.0286	0.391	0.503	
13.0	108	7	0.415	0.0287	0.362	0.475	
14.0	97	8	0.381	0.0288	0.328	0.442	
15.0	78	12	0.322	0.0289	0.270	0.384	
16.0	62	9	0.275	0.0286	0.225	0.338	
17.0	47	6	0.240	0.0283	0.191	0.303	

productiviteit=laag							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	585	131	0.776	0.0172	0.743	0.811	
1.0	444	27	0.729	0.0184	0.694	0.766	
2.0	399	15	0.701	0.0190	0.665	0.740	
3.0	380	17	0.670	0.0197	0.633	0.710	
4.0	353	15	0.642	0.0201	0.603	0.682	
5.0	331	13	0.616	0.0205	0.577	0.658	
6.0	312	16	0.585	0.0209	0.545	0.627	
7.0	287	18	0.548	0.0213	0.508	0.592	
8.0	248	12	0.522	0.0216	0.481	0.566	
9.0	226	13	0.492	0.0219	0.450	0.537	
10.0	206	9	0.470	0.0221	0.429	0.516	
11.0	183	10	0.444	0.0223	0.403	0.490	
12.0	169	16	0.402	0.0226	0.360	0.449	
13.0	133	9	0.375	0.0228	0.333	0.423	
14.0	112	8	0.348	0.0231	0.306	0.397	
15.0	93	7	0.322	0.0234	0.279	0.371	
16.0	80	12	0.274	0.0237	0.231	0.324	
17.0	55	5	0.249	0.0240	0.206	0.301	

productiviteit=medium								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	574	128	0.777	0.0174		0.744		0.812
1.0	438	27	0.729	0.0186		0.694		0.766
2.0	403	19	0.695	0.0193		0.658		0.734
3.0	382	24	0.651	0.0200		0.613		0.692
4.0	356	26	0.604	0.0206		0.564		0.645
5.0	329	15	0.576	0.0209		0.537		0.618
6.0	308	17	0.544	0.0211		0.504		0.587
7.0	279	15	0.515	0.0213		0.475		0.558
8.0	256	19	0.477	0.0214		0.437		0.521
9.0	227	12	0.452	0.0215		0.411		0.496
10.0	207	14	0.421	0.0215		0.381		0.465
11.0	177	10	0.397	0.0216		0.357		0.442
12.0	163	8	0.378	0.0216		0.338		0.423
13.0	145	8	0.357	0.0216		0.317		0.402
14.0	126	14	0.317	0.0217		0.277		0.363
15.0	106	12	0.281	0.0216		0.242		0.327
16.0	78	16	0.224	0.0214		0.185		0.270
17.0	53	9	0.186	0.0212		0.148		0.232
18.0	38	1	0.181	0.0212		0.144		0.227

productiviteit=na								
time	n.risk	n.event	survival	std.err	lower	95% CI	upper	95% CI
0.1	2293	1158	0.4950	0.01044		0.4749		0.5159
1.0	1135	106	0.4488	0.01039		0.4289		0.4696
2.0	1027	93	0.4081	0.01027		0.3885		0.4287
3.0	934	100	0.3644	0.01005		0.3452		0.3847
4.0	829	95	0.3227	0.00977		0.3041		0.3424
5.0	732	89	0.2834	0.00943		0.2655		0.3025
6.0	634	52	0.2602	0.00919		0.2428		0.2788
7.0	461	27	0.2449	0.00911		0.2277		0.2635
8.0	383	21	0.2315	0.00907		0.2144		0.2500
9.0	332	16	0.2204	0.00905		0.2033		0.2388
10.0	300	16	0.2086	0.00903		0.1916		0.2271
11.0	264	24	0.1896	0.00900		0.1728		0.2081
12.0	234	37	0.1597	0.00882		0.1433		0.1779
13.0	190	14	0.1479	0.00872		0.1318		0.1660
14.0	160	14	0.1350	0.00861		0.1191		0.1529
15.0	136	17	0.1181	0.00845		0.1026		0.1359
16.0	106	10	0.1069	0.00836		0.0918		0.1246
17.0	87	20	0.0824	0.00804		0.0680		0.0997

Pairwise comparisons using Log-Rank test

data: wnw and productiviteit

	hoog	laag	medium
laag	0.377	-	-
medium	0.022	0.113	-
na	<2e-16	<2e-16	<2e-16

P value adjustment method: BH

Call:

```
survdiff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ productiviteit,  
data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
productiviteit=hoog	352	221	346	45.2	65.3
productiviteit=laag	585	353	526	57.0	89.6
productiviteit=medium	574	394	522	31.4	49.3
productiviteit=na	2293	1909	1483	122.6	345.1

Chisq= 349 on 3 degrees of freedom, p= <2e-16

Diffusie

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Diffusie, data = wnw)`

Diffusie=BN+NN							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	337	55	0.837	0.0201	0.798	0.877	
1.0	280	17	0.786	0.0224	0.743	0.831	
2.0	243	12	0.747	0.0239	0.702	0.796	
3.0	227	6	0.727	0.0246	0.681	0.777	
4.0	218	15	0.677	0.0261	0.628	0.730	
5.0	201	12	0.637	0.0270	0.586	0.692	
6.0	187	12	0.596	0.0277	0.544	0.653	
7.0	155	7	0.569	0.0283	0.516	0.627	
8.0	130	8	0.534	0.0291	0.480	0.594	
9.0	116	9	0.493	0.0300	0.437	0.555	
10.0	102	5	0.469	0.0304	0.413	0.532	
11.0	86	9	0.419	0.0313	0.362	0.486	
12.0	76	3	0.403	0.0315	0.346	0.470	
13.0	64	5	0.371	0.0320	0.314	0.440	
14.0	55	5	0.338	0.0325	0.280	0.408	
15.0	46	3	0.316	0.0327	0.258	0.387	
16.0	38	2	0.299	0.0331	0.241	0.371	
17.0	30	6	0.239	0.0343	0.181	0.317	

Diffusie=eerst BN dan NN							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
1	360	20	0.944	0.0121	0.921	0.968	
2	339	23	0.880	0.0171	0.847	0.915	
3	316	25	0.811	0.0207	0.771	0.852	
4	290	18	0.760	0.0225	0.717	0.806	
5	270	33	0.667	0.0249	0.620	0.718	
6	234	15	0.625	0.0257	0.576	0.677	
7	190	4	0.612	0.0259	0.563	0.665	
8	172	7	0.587	0.0265	0.537	0.641	
9	161	4	0.572	0.0269	0.522	0.627	
10	154	8	0.542	0.0274	0.491	0.599	
11	138	10	0.503	0.0281	0.451	0.561	
12	120	5	0.482	0.0285	0.429	0.541	
13	110	7	0.451	0.0289	0.398	0.512	
14	92	8	0.412	0.0296	0.358	0.474	
15	74	10	0.356	0.0304	0.302	0.421	
16	56	5	0.325	0.0308	0.270	0.391	
17	40	4	0.292	0.0317	0.236	0.361	

Diffusie=eerst NN dan BN

time	n.risk	n.event	survival	std.err	Lower	95% CI upper	95% CI
1	786	20	0.975	0.00562	0.964	0.986	
2	765	27	0.940	0.00846	0.924	0.957	
3	735	31	0.901	0.01069	0.880	0.922	
4	694	30	0.862	0.01237	0.838	0.886	
5	659	19	0.837	0.01326	0.811	0.863	
6	631	29	0.798	0.01445	0.770	0.827	
7	574	24	0.765	0.01536	0.735	0.796	
8	513	17	0.740	0.01604	0.709	0.772	
9	457	16	0.714	0.01673	0.682	0.747	
10	417	19	0.681	0.01755	0.648	0.716	
11	368	20	0.644	0.01845	0.609	0.681	
12	344	20	0.607	0.01918	0.570	0.645	
13	299	17	0.572	0.01983	0.535	0.612	
14	260	22	0.524	0.02067	0.485	0.566	
15	219	16	0.486	0.02126	0.446	0.529	
16	183	20	0.432	0.02200	0.391	0.478	
17	145	23	0.364	0.02269	0.322	0.411	

Diffusie=enkel BN

time	n.risk	n.event	survival	std.err	Lower	95% CI upper	95% CI
0.1	1113	774	0.3046	0.01380	0.2787	0.3329	
1.0	325	38	0.2690	0.01334	0.2441	0.2964	
2.0	278	25	0.2448	0.01299	0.2206	0.2716	
3.0	252	19	0.2263	0.01268	0.2028	0.2526	
4.0	231	18	0.2087	0.01235	0.1858	0.2344	
5.0	211	34	0.1751	0.01163	0.1537	0.1994	
6.0	174	10	0.1650	0.01139	0.1441	0.1889	
7.0	105	8	0.1524	0.01136	0.1317	0.1764	
8.0	83	8	0.1377	0.01139	0.1171	0.1620	
9.0	74	9	0.1210	0.01129	0.1008	0.1453	
10.0	65	6	0.1098	0.01113	0.0900	0.1339	
11.0	54	3	0.1037	0.01105	0.0842	0.1278	
12.0	48	17	0.0670	0.01011	0.0498	0.0900	
13.0	29	4	0.0577	0.00971	0.0415	0.0803	
14.0	23	1	0.0552	0.00961	0.0393	0.0777	
15.0	21	1	0.0526	0.00951	0.0369	0.0750	
16.0	17	3	0.0433	0.00922	0.0285	0.0657	
17.0	12	3	0.0325	0.00878	0.0191	0.0552	

Diffusie=enkel NN							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	1208	651	0.4611	0.01434	0.43382	0.4901	
1.0	548	85	0.3896	0.01406	0.36297	0.4181	
2.0	460	55	0.3430	0.01371	0.31715	0.3709	
3.0	402	69	0.2841	0.01306	0.25964	0.3109	
4.0	330	65	0.2282	0.01219	0.20547	0.2534	
5.0	264	29	0.2031	0.01171	0.18139	0.2274	
6.0	229	26	0.1800	0.01122	0.15934	0.2034	
7.0	190	21	0.1601	0.01079	0.14033	0.1827	
8.0	165	17	0.1436	0.01039	0.12465	0.1655	
9.0	142	10	0.1335	0.01014	0.11506	0.1550	
10.0	128	11	0.1220	0.00984	0.10421	0.1429	
11.0	114	10	0.1113	0.00954	0.09413	0.1317	
12.0	104	27	0.0824	0.00853	0.06730	0.1010	
13.0	74	5	0.0769	0.00831	0.06218	0.0950	
14.0	65	8	0.0674	0.00793	0.05352	0.0849	
15.0	53	18	0.0445	0.00683	0.03295	0.0601	
16.0	32	17	0.0209	0.00507	0.01296	0.0336	
17.0	15	4	0.0153	0.00441	0.00869	0.0269	
18.0	10	1	0.0138	0.00423	0.00754	0.0251	

Pairwise comparisons using Log-Rank test

data: wnw and Diffusie

	BN+NN	eerst BN dan NN	eerst NN dan BN	enkel BN	enkel NN
eerst BN dan NN	0.0353	-	-	-	-
eerst NN dan BN	8.7e-13	9.8e-07	-	-	-
enkel BN	< 2e-16	< 2e-16	< 2e-16	-	-
enkel NN	< 2e-16	< 2e-16	< 2e-16	0.0093	-

P value adjustment method: BH

Call:

```
survdifff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ Diffusie,  
data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
Diffusie=BN+NN	337	191	298	38.7	55.1
Diffusie=eerst BN dan NN	360	206	369	71.7	103.8
Diffusie=eerst NN dan BN	786	370	916	325.6	649.9
Diffusie=enkel BN	1113	981	579	278.5	525.6
Diffusie=enkel NN	1208	1129	715	240.4	451.3

Chisq= 1462 on 4 degrees of freedom, p= <2e-16

Herkomst

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Leenwoord, data = wnw)`

Leenwoord=inheems							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	1100	261	0.763	0.0128		0.738	0.788
1.0	820	62	0.705	0.0138		0.679	0.733
2.0	733	38	0.669	0.0143		0.641	0.697
3.0	685	40	0.629	0.0147		0.601	0.659
4.0	636	42	0.588	0.0151		0.559	0.618
5.0	589	30	0.558	0.0153		0.529	0.589
6.0	550	29	0.529	0.0154		0.499	0.560
7.0	504	28	0.499	0.0155		0.470	0.531
8.0	457	23	0.474	0.0156		0.444	0.506
9.0	416	20	0.451	0.0157		0.422	0.483
10.0	387	30	0.416	0.0157		0.387	0.448
11.0	334	20	0.391	0.0157		0.362	0.423
12.0	307	30	0.353	0.0157		0.324	0.385
13.0	257	19	0.327	0.0156		0.298	0.359
14.0	223	24	0.292	0.0155		0.263	0.324
15.0	185	26	0.251	0.0153		0.223	0.283
16.0	142	31	0.196	0.0148		0.169	0.227
17.0	98	14	0.168	0.0144		0.142	0.199
18.0	73	1	0.166	0.0144		0.140	0.197

Leenwoord=leenwoord							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	411	61	0.852	0.0175		0.818	0.887
1.0	344	12	0.822	0.0189		0.786	0.860
2.0	325	11	0.794	0.0200		0.756	0.834
3.0	313	10	0.769	0.0209		0.729	0.811
4.0	298	9	0.745	0.0217		0.704	0.789
5.0	284	8	0.724	0.0223		0.682	0.770
6.0	271	11	0.695	0.0231		0.651	0.742
7.0	249	9	0.670	0.0237		0.625	0.718
8.0	223	13	0.631	0.0247		0.584	0.681
9.0	202	12	0.593	0.0255		0.545	0.646
10.0	179	3	0.583	0.0257		0.535	0.636
11.0	162	8	0.555	0.0264		0.505	0.609
12.0	151	5	0.536	0.0268		0.486	0.591
13.0	129	5	0.515	0.0273		0.465	0.572
14.0	112	6	0.488	0.0281		0.436	0.546
15.0	92	5	0.461	0.0289		0.408	0.522
16.0	78	6	0.426	0.0301		0.371	0.489
17.0	57	6	0.381	0.0320		0.323	0.449

Leenwoord=na							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	2293	1158	0.4950	0.01044		0.4749	0.5159
1.0	1135	106	0.4488	0.01039		0.4289	0.4696
2.0	1027	93	0.4081	0.01027		0.3885	0.4287
3.0	934	100	0.3644	0.01005		0.3452	0.3847
4.0	829	95	0.3227	0.00977		0.3041	0.3424
5.0	732	89	0.2834	0.00943		0.2655	0.3025
6.0	634	52	0.2602	0.00919		0.2428	0.2788
7.0	461	27	0.2449	0.00911		0.2277	0.2635
8.0	383	21	0.2315	0.00907		0.2144	0.2500
9.0	332	16	0.2204	0.00905		0.2033	0.2388
10.0	300	16	0.2086	0.00903		0.1916	0.2271
11.0	264	24	0.1896	0.00900		0.1728	0.2081
12.0	234	37	0.1597	0.00882		0.1433	0.1779
13.0	190	14	0.1479	0.00872		0.1318	0.1660
14.0	160	14	0.1350	0.00861		0.1191	0.1529
15.0	136	17	0.1181	0.00845		0.1026	0.1359
16.0	106	10	0.1069	0.00836		0.0918	0.1246
17.0	87	20	0.0824	0.00804		0.0680	0.0997

Call:

```
survdif(formula = Surv(leeftijd, leeft.nog.in.2021) ~ Leenwoord,
data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
Leenwoord=inheems	1100	768	974	43.6	86.3
Leenwoord=leenwoord	411	200	420	115.4	171.9
Leenwoord=na	2293	1909	1483	122.6	345.1

Chisq= 381 on 2 degrees of freedom, p= <2e-16

Pairwise comparisons using Log-Rank test

data: wnw and Leenwoord

	inheems	leenwoord
leenwoord	7.5e-13	-
na	< 2e-16	< 2e-16

P value adjustment method: BH

Domein (een of meerdere)

	een.of.meer.domeinen=na						
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	2752	1263	0.541	0.00950	0.5228	0.560	
1.0	1484	132	0.493	0.00953	0.4746	0.512	
2.0	1343	106	0.454	0.00950	0.4358	0.473	
3.0	1232	112	0.413	0.00940	0.3947	0.432	
4.0	1115	105	0.374	0.00925	0.3562	0.392	
5.0	1006	96	0.338	0.00906	0.3209	0.356	
6.0	897	64	0.314	0.00890	0.2971	0.332	
7.0	706	39	0.297	0.00883	0.2799	0.315	
8.0	602	34	0.280	0.00879	0.2633	0.298	
9.0	530	26	0.266	0.00876	0.2496	0.284	
10.0	480	29	0.250	0.00872	0.2336	0.268	
11.0	415	35	0.229	0.00869	0.2126	0.247	
12.0	370	50	0.198	0.00854	0.1820	0.216	
13.0	312	20	0.185	0.00846	0.1695	0.203	
14.0	261	24	0.168	0.00836	0.1527	0.186	
15.0	213	25	0.149	0.00826	0.1332	0.166	
16.0	171	23	0.129	0.00813	0.1136	0.146	
17.0	135	25	0.105	0.00790	0.0904	0.121	

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ een.of.meer.domeinen, data = wnw)`

een.of.meer.domeinen=een							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	777	168	0.784	0.0148	0.755	0.813	
1.0	595	31	0.743	0.0157	0.713	0.774	
2.0	552	25	0.709	0.0164	0.678	0.742	
3.0	522	32	0.666	0.0171	0.633	0.700	
4.0	481	27	0.628	0.0176	0.595	0.664	
5.0	448	26	0.592	0.0180	0.558	0.628	
6.0	417	24	0.558	0.0182	0.523	0.595	
7.0	378	21	0.527	0.0184	0.492	0.564	
8.0	340	15	0.504	0.0186	0.469	0.541	
9.0	313	14	0.481	0.0187	0.446	0.519	
10.0	292	15	0.456	0.0188	0.421	0.495	
11.0	263	14	0.432	0.0189	0.397	0.471	
12.0	245	17	0.402	0.0189	0.367	0.441	
13.0	204	14	0.375	0.0190	0.339	0.414	
14.0	182	19	0.335	0.0190	0.300	0.375	
15.0	154	15	0.303	0.0189	0.268	0.342	
16.0	124	20	0.254	0.0188	0.220	0.294	
17.0	86	13	0.216	0.0187	0.182	0.256	

een.of.meer.domeinen=meerdere							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	275	49	0.822	0.0231	0.778	0.868	
1.0	220	17	0.758	0.0259	0.709	0.811	
2.0	190	11	0.714	0.0276	0.662	0.771	
3.0	178	6	0.690	0.0284	0.637	0.748	
4.0	167	14	0.632	0.0299	0.576	0.694	
5.0	151	5	0.612	0.0303	0.555	0.674	
6.0	141	4	0.594	0.0307	0.537	0.657	
7.0	130	4	0.576	0.0311	0.518	0.640	
8.0	121	8	0.538	0.0318	0.479	0.604	
9.0	107	8	0.498	0.0325	0.438	0.565	
10.0	94	5	0.471	0.0328	0.411	0.540	
11.0	82	3	0.454	0.0331	0.393	0.524	
12.0	77	5	0.424	0.0335	0.364	0.495	
13.0	60	4	0.396	0.0341	0.335	0.469	
14.0	52	1	0.389	0.0343	0.327	0.462	
15.0	46	8	0.321	0.0357	0.258	0.399	
16.0	31	4	0.280	0.0366	0.216	0.361	
17.0	21	2	0.253	0.0376	0.189	0.339	
18.0	16	1	0.237	0.0385	0.173	0.326	

Pairwise comparisons using Log-Rank test

data: wnw and een.of.meer.domeinen

	een	meerdere
meerdere	0.53	-
na	< 2e-16	2.1e-15

P value adjustment method: BH

Call:

```
survdifff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ een.of.meer.domeinen,  
data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
een.of.meer.domeinen=een	777	510	725	63.8	110.6
een.of.meer.domeinen=meerdere	275	159	242	28.7	39.9
een.of.meer.domeinen=na	2752	2208	1910	46.6	182.0

Chisq= 183 on 2 degrees of freedom, p= <2e-16

Domein ("automatisering, informatie en communicatie")

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ domein, data = wnw)`

domein=automatisering. informatie en communicatie							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	205	31	0.849	0.0250	0.801	0.899	
1.0	169	7	0.814	0.0273	0.762	0.869	
2.0	160	6	0.783	0.0290	0.728	0.842	
3.0	154	8	0.742	0.0308	0.684	0.805	
4.0	146	5	0.717	0.0318	0.657	0.782	
5.0	137	4	0.696	0.0325	0.635	0.763	
6.0	130	9	0.648	0.0340	0.585	0.718	
7.0	116	8	0.603	0.0352	0.538	0.676	
8.0	105	6	0.569	0.0359	0.503	0.644	
9.0	96	6	0.533	0.0364	0.466	0.610	
10.0	88	3	0.515	0.0367	0.448	0.592	
11.0	79	3	0.495	0.0370	0.428	0.573	
12.0	75	5	0.462	0.0373	0.395	0.542	
13.0	59	4	0.431	0.0380	0.363	0.512	
14.0	51	5	0.389	0.0387	0.320	0.472	
15.0	41	3	0.360	0.0392	0.291	0.446	
16.0	31	5	0.302	0.0406	0.232	0.393	
17.0	18	1	0.285	0.0416	0.214	0.380	

domein=na							
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI	
0.1	2752	1263	0.541	0.00950	0.5228	0.560	
1.0	1484	132	0.493	0.00953	0.4746	0.512	
2.0	1343	106	0.454	0.00950	0.4358	0.473	
3.0	1232	112	0.413	0.00940	0.3947	0.432	
4.0	1115	105	0.374	0.00925	0.3562	0.392	
5.0	1006	96	0.338	0.00906	0.3209	0.356	
6.0	897	64	0.314	0.00890	0.2971	0.332	
7.0	706	39	0.297	0.00883	0.2799	0.315	
8.0	602	34	0.280	0.00879	0.2633	0.298	
9.0	530	26	0.266	0.00876	0.2496	0.284	
10.0	480	29	0.250	0.00872	0.2336	0.268	
11.0	415	35	0.229	0.00869	0.2126	0.247	
12.0	370	50	0.198	0.00854	0.1820	0.216	
13.0	312	20	0.185	0.00846	0.1695	0.203	
14.0	261	24	0.168	0.00836	0.1527	0.186	
15.0	213	25	0.149	0.00826	0.1332	0.166	
16.0	171	23	0.129	0.00813	0.1136	0.146	
17.0	135	25	0.105	0.00790	0.0904	0.121	

domein=rest						
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	847	186	0.780	0.0142	0.753	0.809
1.0	646	41	0.731	0.0153	0.702	0.761
2.0	582	30	0.693	0.0160	0.663	0.725
3.0	546	30	0.655	0.0165	0.623	0.688
4.0	502	36	0.608	0.0171	0.576	0.643
5.0	462	27	0.573	0.0174	0.539	0.608
6.0	428	19	0.547	0.0176	0.514	0.583
7.0	392	17	0.523	0.0177	0.490	0.559
8.0	356	17	0.498	0.0179	0.465	0.535
9.0	324	16	0.474	0.0180	0.440	0.511
10.0	298	17	0.447	0.0182	0.413	0.484
11.0	266	14	0.423	0.0183	0.389	0.461
12.0	247	17	0.394	0.0183	0.360	0.432
13.0	205	14	0.367	0.0184	0.333	0.405
14.0	183	15	0.337	0.0185	0.303	0.375
15.0	159	20	0.295	0.0184	0.261	0.333
16.0	124	19	0.250	0.0183	0.216	0.288
17.0	89	14	0.210	0.0182	0.178	0.249
18.0	67	1	0.207	0.0182	0.174	0.246

Pairwise comparisons using Log-Rank test

data: wnw and domein

automatisering. informatie en communicatie na	3.7e-16	-
rest	0.034	< 2e-16

P value adjustment method: BH

Call:

```
survdifff(formula = Surv(leeftijd, leeft.nog.in.2021) ~ domein,
           data = wnw)
```

	N	Observed	Expected	(O-E)^2/E	(O-E)^2/V
domein=automatisering. informatie en communicatie	205	119	201	33.5	45.3
domein=na	2752	2208	1910	46.6	182.0
domein=rest	847	550	766	61.0	108.1

Chisq= 185 on 2 degrees of freedom, p= <2e-16

Synonymie

Call: `survfit(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ heeft.synoniem, data = wnw)`

heeft.synoniem=ja							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	715	110	0.846	0.0135		0.820	0.873
1.0	596	39	0.791	0.0152		0.761	0.821
2.0	540	22	0.759	0.0161		0.728	0.791
3.0	513	20	0.729	0.0168		0.697	0.763
4.0	487	20	0.699	0.0174		0.666	0.734
5.0	462	17	0.673	0.0178		0.639	0.709
6.0	434	22	0.639	0.0183		0.604	0.676
7.0	398	21	0.605	0.0188		0.570	0.643
8.0	366	16	0.579	0.0191		0.543	0.618
9.0	339	17	0.550	0.0194		0.513	0.589
10.0	315	16	0.522	0.0196		0.485	0.562
11.0	273	17	0.490	0.0199		0.452	0.530
12.0	252	15	0.460	0.0201		0.423	0.502
13.0	219	11	0.437	0.0203		0.399	0.479
14.0	194	24	0.383	0.0206		0.345	0.426
15.0	150	16	0.342	0.0207		0.304	0.385
16.0	115	15	0.298	0.0210		0.259	0.342
17.0	86	8	0.270	0.0212		0.231	0.315
18.0	69	1	0.266	0.0213		0.227	0.311

heeft.synoniem=na							
time	n.risk	n.event	survival	std.err	lower	95% CI upper	95% CI
0.1	2293	1158	0.4950	0.01044		0.4749	0.5159
1.0	1135	106	0.4488	0.01039		0.4289	0.4696
2.0	1027	93	0.4081	0.01027		0.3885	0.4287
3.0	934	100	0.3644	0.01005		0.3452	0.3847
4.0	829	95	0.3227	0.00977		0.3041	0.3424
5.0	732	89	0.2834	0.00943		0.2655	0.3025
6.0	634	52	0.2602	0.00919		0.2428	0.2788
7.0	461	27	0.2449	0.00911		0.2277	0.2635
8.0	383	21	0.2315	0.00907		0.2144	0.2500
9.0	332	16	0.2204	0.00905		0.2033	0.2388
10.0	300	16	0.2086	0.00903		0.1916	0.2271
11.0	264	24	0.1896	0.00900		0.1728	0.2081
12.0	234	37	0.1597	0.00882		0.1433	0.1779
13.0	190	14	0.1479	0.00872		0.1318	0.1660
14.0	160	14	0.1350	0.00861		0.1191	0.1529
15.0	136	17	0.1181	0.00845		0.1026	0.1359
16.0	106	10	0.1069	0.00836		0.0918	0.1246
17.0	87	20	0.0824	0.00804		0.0680	0.0997

heeft.synoniem=nee						
time	n.risk	n.event	survival	std.err	lower 95% CI	upper 95% CI
0.1	796	212	0.734	0.0157	0.704	0.765
1.0	568	35	0.688	0.0165	0.657	0.721
2.0	518	27	0.653	0.0170	0.620	0.687
3.0	485	30	0.612	0.0175	0.579	0.647
4.0	447	31	0.570	0.0178	0.536	0.606
5.0	411	21	0.541	0.0180	0.506	0.577
6.0	387	18	0.515	0.0181	0.481	0.552
7.0	355	16	0.492	0.0182	0.458	0.529
8.0	314	20	0.461	0.0184	0.426	0.498
9.0	279	15	0.436	0.0185	0.401	0.474
10.0	251	17	0.407	0.0185	0.372	0.445
11.0	223	11	0.387	0.0186	0.352	0.425
12.0	206	20	0.349	0.0186	0.314	0.387
13.0	167	13	0.322	0.0186	0.287	0.360
14.0	141	6	0.308	0.0186	0.274	0.347
15.0	127	15	0.272	0.0187	0.238	0.311
16.0	105	22	0.215	0.0183	0.182	0.254
17.0	69	12	0.177	0.0180	0.145	0.216

Pairwise comparisons using Log-Rank test

data: wnw and heeft.synoniem

```

      ja      na
na < 2e-16 -
nee 2.9e-06 < 2e-16

```

P value adjustment method: BH

Call:

```
survdif(formula = Surv(leeftijd, leeft.nog.in.2021) ~ heeft.synoniem,
data = wnw)
```

	N	Observed	Expected	(O-E) ² /E	(O-E) ² /V
heeft.synoniem=ja	715	427	705	109.6	188.3
heeft.synoniem=na	2293	1909	1483	122.6	345.1
heeft.synoniem=nee	796	541	689	31.9	54.4

Chisq= 359 on 2 degrees of freedom, p= <2e-16

Appendix V: Cox Proportional Hazards modellen

Samenstelling + Diffusie + Bron

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Samenstelling +
      Diffusie + bron, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
Samenstellingsamenstelling	0.11907	1.12645	0.06630	1.796	0.072503	.
Diffusieeerst BN dan NN	-0.37466	0.68752	0.10123	-3.701	0.000215	***
Diffusieeerst NN dan BN	-0.59318	0.55257	0.08945	-6.631	3.32e-11	***
Diffusieenkel BN	1.06664	2.90559	0.08205	12.999	< 2e-16	***
Diffusieenkel NN	1.10356	3.01487	0.07916	13.940	< 2e-16	***
bronwnw	-0.60752	0.54470	0.04283	-14.186	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Samenstellingsamenstelling	1.1264	0.8877	0.9892	1.2828
Diffusieeerst BN dan NN	0.6875	1.4545	0.5638	0.8384
Diffusieeerst NN dan BN	0.5526	1.8097	0.4637	0.6585
Diffusieenkel BN	2.9056	0.3442	2.4739	3.4126
Diffusieenkel NN	3.0149	0.3317	2.5816	3.5209
bronwnw	0.5447	1.8359	0.5008	0.5924

Concordance= 0.781 (se = 0.005)

Likelihood ratio test= 1737 on 6 df, p=<2e-16

Wald test = 1477 on 6 df, p=<2e-16

Score (logrank) test = 1715 on 6 df, p=<2e-16

Samenstelling + Diffusie + Bron + Lengte

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Samenstelling +
      Diffusie + bron + lengte, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)
Samenstellingsamenstelling	0.08265	1.08616	0.06958	1.188	0.234936
Diffusieeerst BN dan NN	-0.37593	0.68665	0.10122	-3.714	0.000204 ***
Diffusieeerst NN dan BN	-0.59279	0.55278	0.08944	-6.628	3.41e-11 ***
Diffusieeenkel BN	1.06230	2.89303	0.08210	12.939	< 2e-16 ***
Diffusieeenkel NN	1.09922	3.00182	0.07919	13.881	< 2e-16 ***
bronwnw	-0.59505	0.55154	0.04376	-13.598	< 2e-16 ***
lengtelang	0.12226	1.13005	0.07244	1.688	0.091482 .
lengtemedium	0.09878	1.10383	0.06864	1.439	0.150109

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Samenstellingsamenstelling	1.0862	0.9207	0.9477	1.2449
Diffusieeerst BN dan NN	0.6867	1.4563	0.5631	0.8373
Diffusieeerst NN dan BN	0.5528	1.8090	0.4639	0.6587
Diffusieeenkel BN	2.8930	0.3457	2.4630	3.3981
Diffusieeenkel NN	3.0018	0.3331	2.5703	3.5058
bronwnw	0.5515	1.8131	0.5062	0.6009
lengtelang	1.1300	0.8849	0.9805	1.3025
lengtemedium	1.1038	0.9059	0.9649	1.2628

Concordance= 0.783 (se = 0.005)

Likelihood ratio test= 1740 on 8 df, p=<2e-16

Wald test = 1477 on 8 df, p=<2e-16

Score (logrank) test = 1716 on 8 df, p=<2e-16

Samenstelling + Lengte

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Samenstelling +  
      lengte, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
Samenstellingsamenstelling	0.32224	1.38022	0.06954	4.634	3.59e-06	***
lengtelang	0.52718	1.69415	0.07150	7.373	1.67e-13	***
lengtemedium	0.28690	1.33228	0.06869	4.177	2.96e-05	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Samenstellingsamenstelling	1.380	0.7245	1.204	1.582
lengtelang	1.694	0.5903	1.473	1.949
lengtemedium	1.332	0.7506	1.164	1.524

Concordance= 0.573 (se = 0.006)

Likelihood ratio test= 135.3 on 3 df, p=<2e-16

Wald test = 124.6 on 3 df, p=<2e-16

Score (logrank) test = 127.3 on 3 df, p=<2e-16

Bron + Diffusie

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ bron +  
      Diffusie, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
bronwnw	-0.61383	0.54127	0.04270	-14.374	< 2e-16	***
Diffusieeerst BN dan NN	-0.36535	0.69395	0.10112	-3.613	0.000303	***
Diffusieeerst NN dan BN	-0.58435	0.55747	0.08933	-6.542	6.09e-11	***
Diffusieenkel BN	1.08366	2.95549	0.08159	13.282	< 2e-16	***
Diffusieenkel NN	1.11798	3.05866	0.07879	14.189	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
bronwnw	0.5413	1.8475	0.4978	0.5885
Diffusieeerst BN dan NN	0.6940	1.4410	0.5692	0.8461
Diffusieeerst NN dan BN	0.5575	1.7938	0.4679	0.6641
Diffusieenkel BN	2.9555	0.3384	2.5187	3.4680
Diffusieenkel NN	3.0587	0.3269	2.6210	3.5694

Concordance= 0.78 (se = 0.005)

Likelihood ratio test= 1734 on 5 df, p=<2e-16

Wald test = 1478 on 5 df, p=<2e-16

Score (logrank) test = 1714 on 5 df, p=<2e-16

**Samenstelling + Diffusie + Bron + Lengte + Woordsoort + Gewogen
Levenshteinafstand**

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Samenstelling +
      Diffusie + bron + lengte + Woordsoort + gewogenLevenshtein,
      data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)
Samenstellingsamenstelling	0.09190	1.09625	0.07441	1.235	0.216846
Diffusieeerst BN dan NN	-0.37313	0.68858	0.10140	-3.680	0.000234 ***
Diffusieeerst NN dan BN	-0.59378	0.55224	0.08958	-6.629	3.39e-11 ***
Diffusieenkel BN	1.06259	2.89386	0.08221	12.926	< 2e-16 ***
Diffusieenkel NN	1.09661	2.99399	0.07937	13.816	< 2e-16 ***
bronwnw	-0.61452	0.54090	0.04504	-13.645	< 2e-16 ***
lengtelang	0.13618	1.14589	0.07383	1.844	0.065123 .
lengtemedium	0.09096	1.09522	0.06919	1.315	0.188635
Woordsoortoverige	0.25228	1.28695	0.26444	0.954	0.340085
Woordsoortsubstantief	0.06834	1.07073	0.14446	0.473	0.636166
Woordsoortwerkwoord	0.09353	1.09804	0.17992	0.520	0.603185
gewogenLevenshteinlaag	0.03126	1.03175	0.09499	0.329	0.742132
gewogenLevenshteinmedium	0.08041	1.08373	0.04154	1.935	0.052938 .

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Samenstellingsamenstelling	1.0963	0.9122	0.9475	1.2684
Diffusieeerst BN dan NN	0.6886	1.4523	0.5645	0.8400
Diffusieeerst NN dan BN	0.5522	1.8108	0.4633	0.6582
Diffusieenkel BN	2.8939	0.3456	2.4632	3.3998
Diffusieenkel NN	2.9940	0.3340	2.5627	3.4979
bronwnw	0.5409	1.8488	0.4952	0.5908
lengtelang	1.1459	0.8727	0.9915	1.3243
lengtemedium	1.0952	0.9131	0.9563	1.2543
Woordsoortoverige	1.2870	0.7770	0.7664	2.1610
Woordsoortsubstantief	1.0707	0.9339	0.8067	1.4212
Woordsoortwerkwoord	1.0980	0.9107	0.7717	1.5623
gewogenLevenshteinlaag	1.0317	0.9692	0.8565	1.2429
gewogenLevenshteinmedium	1.0837	0.9227	0.9990	1.1757

Concordance= 0.785 (se = 0.005)

Likelihood ratio test= 1745 on 13 df, p=<2e-16

Wald test = 1481 on 13 df, p=<2e-16

Score (logrank) test = 1721 on 13 df, p=<2e-16

Diffusie + Bron + Lengte

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Diffusie +
      bron + lengte, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
Diffusieeerst BN dan NN	-0.37058	0.69034	0.10114	-3.664	0.000248	***
Diffusieeerst NN dan BN	-0.58738	0.55578	0.08933	-6.575	4.86e-11	***
Diffusieeenkel BN	1.07181	2.92065	0.08176	13.110	< 2e-16	***
Diffusieeenkel NN	1.10701	3.02530	0.07894	14.023	< 2e-16	***
bronwnw	-0.59642	0.55078	0.04375	-13.633	< 2e-16	***
lengtelang	0.14890	1.16056	0.06908	2.156	0.031121	*
lengtemedium	0.12154	1.12923	0.06604	1.840	0.065730	.

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Diffusieeerst BN dan NN	0.6903	1.4486	0.5662	0.8417
Diffusieeerst NN dan BN	0.5558	1.7993	0.4665	0.6621
Diffusieeenkel BN	2.9207	0.3424	2.4882	3.4282
Diffusieeenkel NN	3.0253	0.3305	2.5916	3.5315
bronwnw	0.5508	1.8156	0.5055	0.6001
lengtelang	1.1606	0.8617	1.0136	1.3288
lengtemedium	1.1292	0.8856	0.9921	1.2853

Concordance= 0.782 (se = 0.005)

Likelihood ratio test= 1738 on 7 df, p=<2e-16

Wald test = 1478 on 7 df, p=<2e-16

Score (logrank) test = 1716 on 7 df, p=<2e-16

Gewogen Levenshteinafstand + Samenstelling + Woordsoort + Aantal karakters + Bron + Diffusie

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ gewogen +
      Samenstelling + Woordsoort + aantal.karakters + bron + Diffusie,
      data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
gewogen	-0.544435	0.580170	0.215410	-2.527	0.011490	*
Samenstellingsamenstelling	0.099783	1.104931	0.072646	1.374	0.169579	
Woordsoortoverige	0.254741	1.290128	0.264255	0.964	0.335047	
Woordsoortsubstantief	0.070404	1.072941	0.144453	0.487	0.625988	
Woordsoortwerkwoord	0.081431	1.084839	0.179687	0.453	0.650417	
aantal.karakters	0.013741	1.013836	0.005057	2.717	0.006579	**
bronwnw	-0.613843	0.541267	0.045104	-13.609	< 2e-16	***
Diffusieeerst BN dan NN	-0.372611	0.688933	0.101420	-3.674	0.000239	***
Diffusieeerst NN dan BN	-0.591397	0.553554	0.089547	-6.604	3.99e-11	***
Diffusieenkel BN	1.064892	2.900526	0.082234	12.950	< 2e-16	***
Diffusieenkel NN	1.095134	2.989583	0.079388	13.795	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
gewogen	0.5802	1.7236	0.3804	0.8849
Samenstellingsamenstelling	1.1049	0.9050	0.9583	1.2740
Woordsoortoverige	1.2901	0.7751	0.7686	2.1655
Woordsoortsubstantief	1.0729	0.9320	0.8084	1.4241
Woordsoortwerkwoord	1.0848	0.9218	0.7628	1.5428
aantal.karakters	1.0138	0.9864	1.0038	1.0239
bronwnw	0.5413	1.8475	0.4955	0.5913
Diffusieeerst BN dan NN	0.6889	1.4515	0.5647	0.8404
Diffusieeerst NN dan BN	0.5536	1.8065	0.4644	0.6598
Diffusieenkel BN	2.9005	0.3448	2.4688	3.4078
Diffusieenkel NN	2.9896	0.3345	2.5588	3.4929

Concordance= 0.786 (se = 0.005)

Likelihood ratio test= 1748 on 11 df, p=<2e-16

Wald test = 1485 on 11 df, p=<2e-16

Score (logrank) test = 1724 on 11 df, p=<2e-16

Diffusie + Bron + Aantal karakters + Samenstelling

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Diffusie +
      bron + aantal.karakters + Samenstelling, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
Diffusieeerst BN dan NN	-0.377264	0.685735	0.101242	-3.726	0.000194	***
Diffusieeerst NN dan BN	-0.592936	0.552702	0.089448	-6.629	3.38e-11	***
Diffusieeenkel BN	1.060735	2.888494	0.082118	12.917	< 2e-16	***
Diffusieeenkel NN	1.099314	3.002104	0.079194	13.881	< 2e-16	***
bronwnw	-0.590031	0.554310	0.043889	-13.444	< 2e-16	***
aantal.karakters	0.008489	1.008525	0.004553	1.864	0.062260	.
Samenstellingsamenstelling	0.088498	1.092532	0.068288	1.296	0.194993	

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Diffusieeerst BN dan NN	0.6857	1.4583	0.5623	0.8362
Diffusieeerst NN dan BN	0.5527	1.8093	0.4638	0.6586
Diffusieeenkel BN	2.8885	0.3462	2.4591	3.3929
Diffusieeenkel NN	3.0021	0.3331	2.5705	3.5062
bronwnw	0.5543	1.8040	0.5086	0.6041
aantal.karakters	1.0085	0.9915	0.9996	1.0176
Samenstellingsamenstelling	1.0925	0.9153	0.9557	1.2490

Concordance= 0.787 (se = 0.005)

Likelihood ratio test= 1740 on 7 df, p=<2e-16

Wald test = 1480 on 7 df, p=<2e-16

Score (logrank) test = 1718 on 7 df, p=<2e-16

Diffusie + Bron + Aantal karakters

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Diffusie +
      bron + aantal.karakters, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
Diffusieeerst BN dan NN	-0.371311	0.689829	0.101156	-3.671	0.000242	***
Diffusieeerst NN dan BN	-0.586835	0.556084	0.089334	-6.569	5.06e-11	***
Diffusieeenkel BN	1.071393	2.919445	0.081765	13.103	< 2e-16	***
Diffusieeenkel NN	1.108482	3.029756	0.078901	14.049	< 2e-16	***
bronwnw	-0.591402	0.553551	0.043893	-13.474	< 2e-16	***
aantal.karakters	0.009945	1.009995	0.004403	2.258	0.023917	*

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Diffusieeerst BN dan NN	0.6898	1.4496	0.5658	0.8411
Diffusieeerst NN dan BN	0.5561	1.7983	0.4668	0.6625
Diffusieeenkel BN	2.9194	0.3425	2.4872	3.4269
Diffusieeenkel NN	3.0298	0.3301	2.5957	3.5365
bronwnw	0.5536	1.8065	0.5079	0.6033
aantal.karakters	1.0100	0.9901	1.0013	1.0187

Concordance= 0.787 (se = 0.005)

Likelihood ratio test= 1739 on 6 df, p=<2e-16

Wald test = 1481 on 6 df, p=<2e-16

Score (logrank) test = 1718 on 6 df, p=<2e-16

Samenstelling + Aantal karakters

Call:

```
coxph(formula = Surv(leeftijdna2003, leeft.nog.in.2021) ~ Samenstelling +  
      aantal.karakters, data = wnw)
```

n= 3804, number of events= 2877

	coef	exp(coef)	se(coef)	z	Pr(> z)	
Samenstellingsamenstelling	0.331038	1.392413	0.068030	4.866	1.14e-06	***
aantal.karakters	0.037657	1.038375	0.004287	8.783	< 2e-16	***

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

	exp(coef)	exp(-coef)	lower .95	upper .95
Samenstellingsamenstelling	1.392	0.7182	1.219	1.591
aantal.karakters	1.038	0.9630	1.030	1.047

Concordance= 0.584 (se = 0.007)

Likelihood ratio test= 139.8 on 2 df, p=<2e-16

Wald test = 135.2 on 2 df, p=<2e-16

Score (logrank) test = 136.6 on 2 df, p=<2e-16