SPRINT 3

Inhoudsopgave

[5 week 2](#_Toc527392986)

[Cleaning audio/ text data Wikipedia 2](#_Toc527392987)

[Search for new data/ Scraping audio and text data from Uva website 2](#_Toc527392988)

[Mappen structuur Aphasia directory 2](#_Toc527392989)

[6 week 3](#_Toc527392990)

[Corrigeer tekstbestanden waarin losse woorden geen eigen regel hebben 3](#_Toc527392991)

[Alignment zinnen 4](#_Toc527392992)

[Begeleiden met programmeren 5](#_Toc527392993)

# 5 week

|  |
| --- |
| Cleaning audio/ text data Wikipedia |
| Deze week heb ik gewerkt aan de cleaning van de tekst data van de Wikipedia. De cleaning was voornamelijk het weghalen van rare karakters tussen de woorden. Na de cleaning van de tekst data heb ik gekeken of de woorden die worden gesproken in de audio overeen komen met de bijbehorende tekst. Helaas was dit niet het geval. Ook was het probleem dat niet alle woorden een begin en eind tijd hadden. Uit deze reden hebben we voor gekozen om opzoek te gaan naar een andere data bron.  Andere bron, zie tabel ‘*Search for new data/ Scraping audio and text data from Uva website’*. |

|  |
| --- |
| Search for new data/ Scraping audio and text data from Uva website |
| Na een desk research had Erik audio en tekst data bron kunnen vinden die voorheen voor een onderzoek en data verzameling is gebruikt. Zie website: <http://www.fon.hum.uva.nl/>  Om de data van de website op de server te kunnen opslaan heb ik een Scraper Script genaamd ‘*ScrapingDataForSTT.ipynb*’ geschreven. Deze Script realiseert het downloaden van de audio bestanden en de gerelateerde tekst bestanden. De bestanden worden in een gestructureerde manier op de server opgeslagen.  Zie link voor Script:  <https://github.com/troley/aphasia-shared/blob/master/ScrapingDataForSTT.ipynb> |

|  |
| --- |
| Mappen structuur Aphasia directory |
| Deze week heb ik gewerkt aan de mappenstructuur op de server in de directory van Aphasia. Het is altijd handig om een goede structuur te hebben voor het opslaan, verwerken en aanmaken van de data. Dit maakt de paden naar de data overzichtelijk. De basis mappen structuur ziet als volgt uit:    De naam datasource spreekt voor zich, E.G. Uva of VoxForge.org. In het map original staan de originele bestanden opgeslagen dus zoals ze van de source worden ontvangen. Het map transform bestaat uit wavfiles, clean en align. Wavfiles bevat de geconverteerde audiobestanden naar extensie Wav. Hierdoor hoef je niet iedere keer een conversie te uitvoeren, wat invloed heeft op de snelheid van het runnen van je transformatie Script. Clean bevat de tekstbestanden die zijn opgeschoond van ongewenste karakters of ongestructureerde body. Hierdoor zijn ze klaar voor het Aligment proces van de zinnen of gebruikt worden voor een ander doel dat vraagt om een gestructureerde body. In het map Align worden audiosegmenten opgeslagen met daarbij behorende zinnen. E.G. een submap van Align:  Map: audiosegment01 -> audioSentence1.wav, audioSentence2.wav, audioSentence3.wav en alignment.json (Output).  Voor uitleg gestructureerde body zie tabel ‘Alignment zinnen’.  Voor uitleg ‘Output json’ zie tabel ‘Alignment zinnen’.  Link Script voor geautomatiseerd aanmaken van mappen structuur:  <https://github.com/troley/aphasia-shared/blob/master/initialize_directory_structure.ipynb> |

# 6 week

|  |
| --- |
| Corrigeer tekstbestanden waarin losse woorden geen eigen regel hebben |
| Deze week heb ik aantal testen gedaan voor het Alignment proces van de zinnen. Toen vond ik dat hij crashte bij tekstbestanden waarin losse woorden (geen zinnen) achter elkaar staan i.p.v. op een eigen regel.  E.G. ongestructureerde tekstbestand:  AA BB AU OU  1 2 3 4  En het moet zijn:  AA  BB  AU  OU  1  2  3  4  Vervolgens heb ik een ander fout gevonden. De cijfers staan als nummers genoteerd en niet als woord. Dus 2 staat als 2 maar niet als twee.  Om deze problemen te kunnen corrigeren heb ik een methode voor geschreven. Deze methode haalt alle tekstbestanden op die deze problemen hebben en geeft ze een gestructureerde body voor het Alignment proces. Ook veranderd hij de cijfers naar woorden.  Link Script:  <https://github.com/troley/aphasia-shared/blob/master/SeparateCharfiles.ipynb> |

|  |
| --- |
| Alignment zinnen |
| Deze week heb ik desk research gedaan naar bibliotheken voor alignment van zinnen met de audiosegmenten. Daarna heb ik een Script geschreven die de begin en eindtijd terug geeft van wanneer een zin begint en eindigt in een audio.  Voor dit doel heb ik het bibliotheek Aeneas gevonden. Hij staat heel goed aangeschreven (<https://github.com/pettarin/forced-alignment-tools> ).    Source: <https://github.com/pettarin/forced-alignment-tools>  De Script knipt vervolgens de audio in segmenten die behoort tot een zin. Vervolgens wordt een map aangemaakt met de naam van de audio en daarin worden de audiosegmenten opgeslagen met de ‘OUTPUT’ json bestand.  In de OUTPUT json bestand staan de begin, eindtijd en de zin opgeslagen van de gehele audio.  De Script genereert vervolgens een CSV of tekstbestand waarin op elk regel een zin staat met daarnaast de gerelateerde audiosegment pad. Dit is nodig voor Sphinx om hem te kunnen trainen. Het bestand wordt opgeslagen in het submap ‘final’ van het map ‘Uva’ op de server.  Voordat de script mag uitgevoerd worden moeten de tekstbestanden in het map Clean de juiste gestructureerde body hebben. De gestructureerde body heeft op elk regel een zin en de eerste regel in het bestand heeft altijd de cijfer 1. Door op de eerste regel de cijfer 1 te hebben kun je de stilte in het begin overslaan. Anders krijg je ook de stilte mee in de eerste audiosegment opgeslagen.  Link Script:  <https://github.com/troley/aphasia-shared/blob/master/aligner.ipynb>  Link Aeneas:  <https://github.com/readbeyond/aeneas>  <https://www.readbeyond.it/aeneas/docs/clitutorial.html> |

|  |
| --- |
| Begeleiden met programmeren |
| Deze week heb ik mijn projectgenoot Jesse begeleidt bij het implementeren van een Script. |