

SPC 2011: Luisterexperiment

David Weenink

6 juni 2011

Samenvatting

Dit experiment betreft het randgebied van “phonemic restoration”. In navolging van Shriberg [1992]¹ gaan we een experiment uitvoeren met gefilterde klinkers. Dat betekent opnames maken, klinkermateriaal selecteren, bestandsnamen systematiseren, experiment opzetten, experiment uitvoeren en gegevens verwerken, conclusies trekken en een verslag maken in artikelvorm waarin alle bovenstaande aspecten worden behandeld.

1 Inleiding

De bedoeling van dit document is dat het een leidraad vormt voor het voorbereiden, het uitvoeren en het verwerken van een variant van het experiment van Shriberg [1992] met Nederlandse klinkers. Dit document gaat er van uit dat je het Shriberg experiment kent. We gaan ons experiment gezamenlijk doen. Dit betekent dat iedere student zijn eigen klinkermateriaal gaat opnemen in de studio. Vervolgens gaat ieder met zijn eigen klinkers aan de slag om hiervan stimulusmateriaal te maken. Het stimulusmateriaal van alle studenten wordt verzameld en hieruit gaan we dan het materiaal van een aantal sprekers selecteren om als stimuli voor het luisterexperiment te fungeren (het experiment zou te lang duren als we het materiaal van alle studenten zouden gebruiken). Iedere student maakt dan met *hetzelfde* stimulusmateriaal een luisterexperiment waarbij de volgorde van de stimuli telkens anders is. Vervolgens plannen we een uur waarin we gezamenlijk het luisterexperiment

¹Zie de URL: http://www.fon.hum.uva.nl/david/ba_spc/2011/shriberg-1992.pdf.

doen. De responses van het experiment kunnen dan verzameld worden. Als eindopdracht krijgt iedereen dan de kans om een verslag van het experiment te maken in artikelvorm.

Het mooie van dit experiment is dat we alle stimulusmateriaal geheel geautomatiseerd kunnen maken uit de basisbestanden waarin onze klinkers zitten. Als we bestanden met klinkermateriaal hebben dan is het mogelijk om met deze bestanden via scripts het hele experiment te maken. We zullen hiervoor alle scripttechnieken die we geleerd hebben moeten aanspreken.

2 Bestandsnamen en structuur

Om zo efficiënt mogelijk te kunnen scripten is het handig dat we systeem aanbrengen in alle bewerkingen die plaats moeten vinden. Dit begint met het opzetten van een directorystructuur zoals we in figuur 1 kunnen zien. We maken een directory `exp2011` op de USB-stick met daarin een aantal subdirectories die bepaalde informatie gaan bevatten. Deze subdirectories zijn:

raw Bevat de ruwe onbewerkte studio-opnames en de TextGrids die de klinker annoteren. Uit deze map mag nadat de relevante bestanden erin staan alleen nog maar gelezen worden. Alle stimuli voor het experiment moeten uit deze map via scripts geproduceerd kunnen worden.

mystim Bevat de scripts om je eigen stimuli te maken uit het materiaal in `raw`. De eigen stimuli komen ook als `wav`-bestand in deze map.

stim Bevat alleen de stimuli die voor het experiment nodig zijn (de `nx60` stimuli van de gekozen studenten).

exp De scripts om het experiment te verwerken.

responses Bevat alleen de bestanden die de antwoorden van alle luisteraars bevatten.

analysis De werkmap met analysegegevens van de stimuli.

Omdat we met klinkerbestanden werken waarvan we de inhoud willen gebruiken en verder bewerken is het handig om de bestandsnaamgeving zo aan te passen dat we uit de naam de informatie kunnen halen die uiteindelijk van

belang is. Omdat we voor het luisterexperiment verschillende *klinkers* van verschillende *sprekers* onder verschillende *condities* willen gebruiken is het handig deze drie variabelen *in de bestandsnamen* te coderen. Dit betekent dat we aan de bestandsnaam kunnen zien welke klinker er in zit, door wie de klinker is uitgesproken en welke bewerking de klinker heeft ondergaan. Hiervoor moeten we allereerst onze opnamebestanden aanpassen. Omdat we voor het experiment alleen geïnteresseerd zijn in de twaalf Nederlandse klinkers gaan we alleen verder met de zinnestjes van de vorm “In sVs en in sVsse zit de V” en “In sVs en in sVze zit de V”. Zorg ervoor dat elk zinnestje in een aparte file komt en geef die file dan een naam die afgeleid is van `Snn_sVs_tt.wav`, waarbij “`tt`” één van de symbolen uit het rijtje van tabel 2 is en “`nn`” de *tweecijferige* sprekercodering die je in tabel 1 kunt vinden. Let op het verschil tussen kleine letters en hoofdletters!

De stimuli voor het luisterexperiment krijgen namen waarin behalve de spreker en de klinker ook het soort bewerking dat de stimulus heeft ondergaan is af te leiden: `Snn_Vtt_Ck.wav`, waarin `nn` de sprekercode is, `tt` de klinkercode en `k` de experimentele conditie. De bewerkingen staan in paragraaf 3.3.

Bijvoorbeeld voor klinker *ie* van spreker *02* in de conditie *LPN/HPF* is de bestandsnaam `S02_Vie_C4.wav`. Let weer op de kleine letters en de hoofdletters!

(..)

exp2011

raw mystim stim exp responses analysis

Figuur 1: De directoryhiërarchie voor het experiment.

3 Het spraakmateriaal

De twaalf Nederlandse klinkers uit de voorgelezen zinnestjes “In sVs en in sVsse zit de V” en “In sVs en in sVze zit de V” worden gebruikt om als basis te dienen voor het experiment.

3.1 Selectie van de klinkers en klinkerduurbepaling

Nadat alle 12 klinkers in het zinnetje in de `raw` directory zitten gaan we de klinkers annoteren (opdracht 7). Markeer begin- en eindtijd van de klinkers in `sVs`, `sVsse`, `sVze` en in `V` op *nuldoorgangen*. Neem alleen stemhebbende delen van de klinker mee. Zet de eindgrens zo dat de intensiteit bij de eindgrens niet meer dan 15 dB onder de maximale intensiteit van de klinker ligt. Label de drie klinkers in elke zin in één intervaltier met de codes zoals weergegeven in tabel 2. Dit is weer nodig om de latere verwerking beter te kunnen automatiseren. Voor het uiteindelijke luisterexperiment gebruiken we de klinker in de `s_sse` of de `s_ze` context.

Bepaal de klinkerduren van je 12 klinkers met een script. De resultaten komen in een tabel te staan met elf kolommen (spreker, klinker, begintijd, eindtijd, duur, ...).

3.2 Formantfrequenties bepalen

Iedere student meet van zijn eigen klinkers op 40 ms vanaf het begin van elke klinker de eerste drie formantfrequenties en de toonhoogte en zet deze in een Table. Je bepaalt ook klinkerduur. Je kunt de Table maken via `Create Table with column names... Snn 12 speaker vowel F0 F1 F2 F3 duration`, waarbij in “Snn”, de “nn” jouw spreker nummer is. De formantfrequenties bepaal je met het commando `To Formant (burg)...`. Kies voor **Maximum formant** de waarde 5500 Hz als je een vrouw bent en 5000 Hz als je een man bent en voor **Max. number of formants** altijd 5. Uit het Formant-object kun je daarna via de opties onder Query de formantfrequenties halen voor het interval van de klinker waarvan je ze wilt weten.

De toonhoogte kun je meten door eerst het commando `To Pitch...` uit te voeren en daarna een query van het Pitch-object te doen.

3.3 Bewerkingen

- Vier verschillende stimuli. LP (low pass): Laagdoorlaat gefilterd, filteramplitude is 1 bij 700 Hz en 0 bij 1000 Hz. HP (high pass) is hoogdoorlaat (eigenlijk bandgefilterd), filteramplitude is 0 bij 1000 Hz en 1 vanaf 1300 Hz tot 10000 Hz, en 0 vanaf 10300 Hz.

1. C1: Ongemodificeerde klinkers 0-10 kHz, 100 ms duur.

2. C2: LPV. Klinkers laagdoorlaat gefilterd.
 3. C3: LPV/HPN. Klinkers laagdoorlaat gefilterd in 900 ms ruis; de klinker begint bij 400 ms. De ruis is complementair en `randomUniform(-0.5,0.5)` hoogdoorlaatgefilterd.
 4. C4: LPN/HPV. Laagdoorlaat gefilterde ruis met hoogdoorlaat gefilterde klinkers.
 5. C5: HPV. Hoogdoorlaat gefilterde klinkers.
- Duurnormalisatie. Alle klinkers moeten dezelfde duur krijgen (100 ms) (`Sound:Extract part... 0 duur rectangular 1 no`)
 - Fade-in en fade-out gedurende 10 ms.
 - Amplitudenormalisatie door eerst de maximale amplitude van een klinker op 0.5 te zetten (`Sound: Scale peak... 0.5`)
 - Optellen van klinker en witte ruis (`randomUniform (-0.5,0.5)`).

4 Experiment uitvoeren

Voor het uitvoeren van experiment moeten in het algemeen een aantal keuzes gemaakt worden.

4.1 Hoeveel stimuli

Het aantal stimuli dat we aan willen of aan kunnen bieden in een experiment hangt af een aantal factoren. Factoren zijn o.a. beschikbaarheid van voldoende goede opnames, beschikbaarheid van proefpersonen (pp), beschikbaarheid van experimenteerruimte etc. De statistische betrouwbaarheid die we willen stelt ook voorwaardes aan het minimale aantalproefpersonen die we nodig hebben. We willen dit experiment kunnen doen gedurende één les van ongeveer vijfenveertig minuten (dit zijn ongeveer 2700 secondes). Onze stimuli duren maximaal 0.9 s. Als we ongeveer vier of vijf secondes schatten voor de tijd die een pp nodig heeft om een stimulus te horen, deze te verwerken en een responsie te geven dan kunnen we uitrekenen dat we zo tussen de 540 ($=2700/5$) en 675 ($=2700/4$) stimuli kunnen aanbieden. We hebben dan nog geen rekening gehouden met noodzakelijke tussentijdse pauzes omdat pp

over het algemeen geen vijfenveertig minuten aaneengesloten geconcentreerd kunnen blijven.

Gegeven deze grenzen aan het aantal stimuli kunnen we stellen dat het aantal spekers die we kunnen gebruiken maximaal elf is (elf sprekers leveren elf maal zestig is zeshonderzestig stimuli). Wanneer we dit aantal sprekers willen gebruiken dan kunnen we het materiaal van elke spreker maar één keer aanbieden aan de pp. Elke stimulus wordt dan precies één keer gebruikt in het experiment.

4.2 Hoe bieden we de stimuli aan

Om voor iedereen dezelfde luistercondities te kunnen garanderen moeten we de stimuli aanbieden via een koptelefoon.

4.3 Wat voor type experiment

We willen dat een proefpersoon elke klank die ze gehoord heeft benoemd. In ons experiment willen we dat de proefpersoon kiezen uit een vast aantal voorgeschreven mogelijkheden. Zo'n experiment heet een *gedwongenkeuze-experiment*. Met behulp van Praat kunnen we zo'n luisterexperiment uitvoeren. We moeten dan een tekstbestand maken waarin de beschrijving van het experiment staat. Hoe dit bestand er precies uit moet zien kun je zien de de Help>ExperimentMFC >The experiment file. Uit de hoeveelheid regels die dit bestand beslaat kun je zien dat er nog al wat komt kijken bij zo'n luisterexperiment.

4.4 Hoe moet de proefpersoon responsies geven

Omdat we het experiment via een computer met Praat uitvoeren liggen als responsies eigenlijk alleen ingetikte tekst of aanwijzen met een muis voor de hand. Andere responsies zoals via spraak of via invullen van een scoreformulier zijn dan minder geschikt. We hebben hier alleen keuzes uit een vaste set en daarvoor is aanklikken via een muis het meest geschikt.

We hebben de volgende layout voor de responsies op het scherm:

eu uu ee ie aa oo oe

u e i a o

Deze layout is gekozen om zo duidelijk mogelijk de klinker te kunnen selecteren. Om verwarring, door het schriftbeeld, tussen lange klinkers en de korte klinkers te minimaliseren zijn de lange klinkers boven de korte geplaatst. Bovendien staat elke lange klinker zo dicht mogelijk bij de korte klinker waarmee hij op grond van zijn spelling mee verward zou kunnen worden.

4.5 Welke instructie geven we de proefpersoon

De pp moet een korte instructie krijgen over wat zij te horen krijgt en wat ze geacht wordt te doen tijdens het experiment. Leg uit welke klanken bedoeld worden met de opschriften. Meestal zijn er ook wat stimuli aan het begin om te oefenen. In het script in paragraaf ?? hebben we ook rekening gehouden met oefenstimuli en lassen we een pause in als er een bepaalde hoeveelheid stimuli afgeluisterd zijn.

4.6 Wanneer voeren we het experiment uit

Via `Read from file...` lees je het experiment-bestand in. Er verschijnt een nieuw `experimentMFC`-object. Kies nu de optie “Run” en het experiment start met een leeg veld met in het midden de tekst “Klik om te beginnen.” Wees er op bedacht dat na de klik onmiddellijk de eerste stimulus volgt. Na de klik op één van de twaalf klinkers volgt snel hierna de volgende stimulus... Na tweehonderd stimuli is er de mogelijkheid om een (korte) pauze te nemen als de tekst “Korte pauze. Klik om verder te gaan.”. Na de klik volgt onmiddellijk weer de volgende stimulus...

Als uiteindelijk de tekst "Einde van het experiment" verschijnt dan kun je het experimentvenster sluiten door `File>Close` te kiezen. Kies nu de optie `Extract results` en hierna verschijnt een nieuw object van het type `ResultsMFC`. Schrijf dit resultaat onmiddellijk weg onder de naam `Pnn.ResultsMFC`, in de directory `responses` waarbij `nn` je tweecijferige nummer is. Upload het naar BlackBoard.

5 Gegevens verwerken en uitwisselen

Uitwisseling van bestanden gaat via BlackBoard>Users and Groups>Groups >10.OTLA... >File Exchange. Hier kan iedereen zijn geluiden en gegevens kwijt.

5.1 Stimuli uploaden

Om het overzichtelijk te houden moet iedereen zijn audiostimuli (60 stuks) via één containerbestand, een zogenaamde Collection, opsturen. Deze Collection moet je de naam `Snn_V60.Collection` geven, waarbij uiteraard `nn` staat voor je tweecijferige sprekerencoding. De snelste manier om een Collectionbestand te maken is om precies alle 60 stimuli als `sound` in het objectvenster aanwezig te hebben, ze dan alle 60 samen te selecteren en dan `Write>Write to binary file...` te kiezen.

5.2 Resultaten verwerken

De resultaten van het luisterexperiment staan per proefpersoon in een apart bestand `Pnn.ResultsMFC`. Deze bestanden moet je allemaal overhalen van BB en in je responses directory zetten. Je kunt ze dan allemaal tegelijk inlezen in Praat, samen selecteren en vervolgens `Collect to Table` kiezen. Je krijgt dan één grote tabel met alle resultaten erin. Bewaar deze tabel in je responses directory. Vervolgens gaan we deze tabel wat aanpassen om er gemakkelijk de verwarringsmatrices uit te kunnen halen. We verwijderen eerst de `reactionTime` kolom (`Table > Modify > Remove column..`). We houden dan een tabel over met drie kolommen. In de stimulus kolom staan de `Snn_Vtt_Ci` namen. In deze namen staan drie types informatie die we elk in een aparte kolom gaan zetten. Dit is makkelijker voor het vervolg. We voegen drie nieuwe kolommen toe aan de tabel: `speaker`, `vowel` en `condition`. De `speaker` kolom gaat als volgt:

```
|| Insert column... 4 speaker  
|| Formula (column range)... speaker speaker mid$(self$[row,"stimulus"],1,2)
```

We halen uit de string in de `stimulus` kolom een substring die we in de `speaker` kolom zetten (let op: we gebruiken hier `self$` i.p.v. `self` omdat we met een string te maken hebben). Op een analoge manier kunnen we de twee tekens die de klinker representeren in de nieuwe `vowel` kolom zetten en het ene teken voor de conditie in de `condition` kolom.

5.3 Verwarringsmatrices maken

(Vanaf Praat versie 5.2.26!) In praat is een verwarringsmatrix een object van het type Confusion. Je kunt een (lege) Confusion maken via het `New > Tables > Create simple Confusion...` commando. Een cell in een Confusion wordt geïndexeerd door een stimuluscategorie (rijlabel) en een responscategorie (kolomlabel). De inhoud van cel met stimuluslabel "oe" en responslabel "aa" geeft aan hoe vaak stimulus "oe" de respons "aa" gekregen heeft. Nuttige commandos met een Confusion:

Create simple Confusion... maakt een verwarringsmatrix waarbij de stimulus en responscategorieën gelijk zijn aan de opgegeven labels. Door onze twaalf klinkercategorieën bij het tweede argument van dit commando op te geven kunnen we een twaalf bij twaalf verwarringsmatrix maken.

Increase... verhoogt het getal in de cell die geïndexeerd wordt door een stimulus en een respons met 1. Met dit commando vul je een Confusion. Je leest uit de tabel met responses van het luisterexperiment telkens een nieuwe regel een stimulus en de respons en met dit commando kun je dan de inhoud van de corresponderende cell in de Confusion aanpassen.

Group... groepeert stimuli en responses onder een nieuw label. Het commando `Group... "oe oo oh ah" B 0` groepeert de achterklinkers onder het label B. De nieuwe verwarringsmatrix heeft nu drie rijen en drie kolommen minder. Met `Group... "ie uu ee ih uh eu eh aa" F 0` groeperen we de voorklinkers. Voor het gemak hebben we de ih en de uh onder de voorklinkers geplaatst. Als je de ih en de uh niet bij de voorcategorie wilt plaatsen kun je nog een derde categorie O (other) definiëren waarin je alles zet dat geen F of B is.

Zoals boven beschreven hebben we dus *twee* acties nodig om een complete twaalf bij twaalf verwarringsmatrix naar een twee bij twee matrix met categorieën F en B te transformeren: eerst om de voorklinkers te groeperen onder het F label en dan om de achterklinkers te groeperen onder het B label.

6 Verslag schrijven

Het verslag begint met de titel, auteur, datum. Dan een korte samenvatting, gevolgd door een inleiding die het experiment kort beschrijft. Dan een sectie

over het spraakmateriaal hoe het is verzameld en verwerkt. Dan beschrijving hoe het experiment is gedaan, welke proefpersonen en hoeveel, de luistercondities. Dan een stuk over de uitkomsten van het experiment, dat wordt gevolgd door een discussie/conclusie. Gebruikte literatuur. Een appendix met de scripts die je hebt gebruikt. Je kunt het verslag van het experiment zoals beschreven in Shriberg [1992] als leidraad nemen.

7 Coderingen

Tabel 1: De sprekercoderingen

Naam	Code
Iris Broedelet	01
Samantha Buijnk	02
Marlon de Groot	03
Liesbeth Holstvoogd	04
Joos Jonges	05
Mala Joosten	06
Imke Kruitwagen	07
Tessa Meijer	08
Anne Meijnderts	09
Annelieke Roeleveld	10
Sheean Spoel	11
Mart van Baalen	12
Luuk van de Scheur	13
Charlotte van den Berg	14
Marieke van den Heuvel	15
Max van Nieukerken	16
Carmen Lie-Lahuerta	17
Ginny Abbink	18
Cassandra Nijon	19
Nienke van der Veen	20
Kees van Doorn	21

Tabel 2: De klinkercoderingen met twee tekens.

IPA	tt
u	oe
o:	oo
a:	aa
e:	ee
i	ie
y	uu
ø:	eu
ɔ	oh
ɑ	ah
ɛ	eh
ɪ	ih
ʏ	uh

Referenties

Elisabeth E. Shriberg. Perceptual restoration of filtered vowels with noise.
Language and Speech, 35:127–136, 1992.