Final assessment part 3.1: Data transformation for comparison between Dutch and English speakers using the ASA Questionnaire

Johan Hensman

21 June, 2023

Contents

1 Introduction

This document presents the data transformation needed for the final assessment question 3: Comparison between Chinese and Dutch speakers on their perception of ASAs as reported in the paper Code based on Fengxiang Li with adaptations made to Dutch.

We use the following packages:

```
library(readxl)   # Read Excel file
library(foreign)  # Open various data files
library(haven)    # Use read_sav function
```

```
library(dplyr)    # Use select function
library(reshape2) # Use melt function
library(knitr)    # Get markdown file
library(tinytex)  # Use TeX environment
library(rticles)  # Use CTeX documents template
library(pander) # For pandering tables
panderOptions("table.alignment.default","left")
```

2 Data files

The input data for this analysis is based on 3 files. These aren't the fully raw data collected on Qualtrics as it pre-processed to anonymize and delte columns that are not needed/allowed to publish.

2.1 Final_ASA_Dutch_Round_1_First_Half_anonym.sav"

The first survey was split in two to prevent fatigue. In the first group first 12 constructs were asked and the second group the last 12 constructs. The participants rated 44 and 46 questionnaire items each and 14 attention checks that all had to be correct. The participants had the option to not recommend their data, but this was of importance when creating the questionnaire. In this final assessment all data will be used. this file contains data of the first 12 constructs

2.2 Final_ASA_Dutch_Round_1_Second_Half_anonym.sav

this file contains data from the last 12 constructs

2.3 ASA_Chinese_Round1_Part1.xlsx and _Part2.xlsx

In the same way that the Dutch survey was split in half, the same happened to the Chinese study where the first half contains data on the first 12 constructs and the second part contains the latter 12 constructs of the questionnaire

```
exceldata0 = read_excel("ASA_Chinese_Round1_Part1.xlsx")
dd0 = data.frame(exceldata0)
exceldata1 = read_excel("ASA_Chinese_Round1_Part2.xlsx")
dd1 = data.frame(exceldata1)
dd2 <- read_sav("Final_ASA_Dutch_Round_1_First_Half_anonym.sav")
dd3 <- read_sav("Final_ASA_Dutch_Round_1_Second_Half_anonym.sav")
```

3 Transformation of the data

3.1 Transforming Chinese data

The original questionnaire had 131 items and the newer version, which was used in the Dutch study, only had 90 items. The 41 items that were not needed anymore were searched and omitted from the data.

```
dd0 <- dd0[-c(5,19,20,23,24,29,30,31,33,34,36,37,39,40,46,50,53,54,58,60,63)]
dd1 <- dd1[-c(2,4,6,7,9,12,17,18,21,25,26,28,35,39,40,42,45,49,60,62)]
```

The 2 datasets are then merged into 1 to ease the calculations.

```
dd4 <- merge(data.frame(dd0), data.frame(dd1),
  by = 0, all = TRUE)[-1]
```

The column names names of the original data are hard to read, so we changed them to the tags of the original English ASA questionnaire.

```
colnames(dd4)[1:90] <- c("HLA1", "HLA2", "HLA3", "HLA4", "HLB1", "HLB2", "HLB3",
                         "HLB4", "HLB5", "NA1", "NA2", "NA3", "NA4", "NA5",
                         "NB1", "NB2", "NB3", "AAS1", "AAS2", "AAS3", "AU1",
                         "AU2", "AU3", "PF1", "PF2", "PF3", "AL1", "AL2",
                         "R_AL3", "AL4", "AL5", "AS1", "AS2", "AS3", "APP1",
                         "R_APP2", "APP3", "UAA1", "UAA2", "R_UAA3", "R_AE1",
                         "AE2", "AE3", "R_AE4", "UE1", "UE2", "UE3", "UT1",
                         "UT2", "UT3", "UAL1", "UAL2", "UAL3", "UAL4", "UAL5",
                         "UAL6", "AA1", "AA2", "AA3", "R_AC1", "R_AC2", "R_AC3",
                         "R_AC4", "AI1", "AI2", "R_AI3", "AI4", "AT1", "AT2",
                         "R_AT3", "SP1", "SP2", "SP3", "IIS1", "IIS2", "IIS3",
                         "IIS4", "AEI1", "AEI2", "R_AEI3", "AEI4", "R_AEI5",
                         "UEP1", "UEP2", "UEP3", "UEP4", "UAI1", "UAI2", "UAI3",
                         "UAI4")
```

As the data is from [1,7] we transform it to [-3,3] like the Dutch data and reverse the scores of the items that start with R_.

```
for (j in 1:90){
  dd4[[j]][] <- dd4[[j]][]-4 # Transform scores from the range [1,7] into [-3,3]
}
```

```
for (i in grep("R_",colnames(dd4))){
# Find column number of reversing-scoring items and translations
  dd4[[i]][] <- dd4[[i]][]*(-1)
  # Reverse scores of reverse-scoring items and translations
}
```

For each construct available we select the columns that belong to their construct and store them in a separate dataframe to be able to calculate the average scores

```
d1_bilingual <-dd4
HLA_CH<-data.frame(select(d1_bilingual,HLA1:HLA4))
HLB_CH<-data.frame(select(d1_bilingual,HLB1:HLB5))
NA_CH<-data.frame(select(d1_bilingual,NA1:NA5))
NB_CH<-data.frame(select(d1_bilingual,NB1:NB3))
AAS_CH<-data.frame(select(d1_bilingual,AAS1:AAS3))
AU_CH<-data.frame(select(d1_bilingual,AU1:AU2,AU3))
PF_CH<-data.frame(select(d1_bilingual,PF1:PF3))
AL_CH<-data.frame(select(d1_bilingual,AL1:AL5))
AS_CH<-data.frame(select(d1_bilingual,AS1:AS3))
APP_CH<-data.frame(select(d1_bilingual,APP1:APP3))
UAA_CH<-data.frame(select(d1_bilingual,UAA1:R_UAA3))
AE_CH<-data.frame(select(d1_bilingual,R_AE1:R_AE4))
UE_CH<-data.frame(select(d1_bilingual,UE1:UE3))
UT_CH<-data.frame(select(d1_bilingual,UT1:UT3))
UAL_CH<-data.frame(select(d1_bilingual,UAL1:UAL6))
AA_CH<-data.frame(select(d1_bilingual,AA1:AA3))
AC_CH<-data.frame(select(d1_bilingual,R_AC1:R_AC4))
AI_CH<-data.frame(select(d1_bilingual,AI1:AI4))
AT_CH<-data.frame(select(d1_bilingual,AT1:R_AT3))
SP_CH<-data.frame(select(d1_bilingual,SP1:SP3))
IIS_CH<-data.frame(select(d1_bilingual,IIS1:IIS4))
AEI_CH<-data.frame(select(d1_bilingual,AEI1:R_AEI5))
UEP_CH<-data.frame(select(d1_bilingual,UEP1:UEP4))
UAI_CH<-data.frame(select(d1_bilingual,UAI1:UAI4))
# Select English item scores of each construct/dimension

d1_bilingual$HLA_CH<-rowMeans(HLA_CH, na.rm=TRUE)
```

```
d1_bilingual$HLB_CH<-rowMeans(HLB_CH, na.rm=TRUE)
d1_bilingual$NA_CH<-rowMeans(NA_CH, na.rm=TRUE)
d1_bilingual$NB_CH<-rowMeans(NB_CH, na.rm=TRUE)
d1_bilingual$AAS_CH<-rowMeans(AAS_CH, na.rm=TRUE)
d1_bilingual$AU_CH<-rowMeans(AU_CH, na.rm=TRUE)
d1_bilingual$PF_CH<-rowMeans(PF_CH, na.rm=TRUE)
d1_bilingual$AL_CH<-rowMeans(AL_CH, na.rm=TRUE)
d1_bilingual$AS_CH<-rowMeans(AS_CH, na.rm=TRUE)
d1_bilingual$APP_CH<-rowMeans(APP_CH, na.rm=TRUE)
d1_bilingual$UAA_CH<-rowMeans(UAA_CH, na.rm=TRUE)
d1_bilingual$AE_CH<-rowMeans(AE_CH, na.rm=TRUE)
d1_bilingual$UE_CH<-rowMeans(UE_CH, na.rm=TRUE)
d1_bilingual$UT_CH<-rowMeans(UT_CH, na.rm=TRUE)
d1_bilingual$UAL_CH<-rowMeans(UAL_CH, na.rm=TRUE)
d1_bilingual$AA_CH<-rowMeans(AA_CH, na.rm=TRUE)
d1_bilingual$AC_CH<-rowMeans(AC_CH, na.rm=TRUE)
d1_bilingual$AI_CH<-rowMeans(AI_CH, na.rm=TRUE)
d1_bilingual$AT_CH<-rowMeans(AT_CH, na.rm=TRUE)
d1_bilingual$SP_CH<-rowMeans(SP_CH, na.rm=TRUE)
d1_bilingual$IIS_CH<-rowMeans(IIS_CH, na.rm=TRUE)
d1_bilingual$AEI_CH<-rowMeans(AEI_CH, na.rm=TRUE)
d1_bilingual$UEP_CH<-rowMeans(UEP_CH, na.rm=TRUE)
d1_bilingual$UAI_CH<-rowMeans(UAI_CH, na.rm=TRUE)
# Calculate English mean of each construct/dimension per participant
```

We added the culture tag 0 for the Chinese data and the ConstructID for each construct.

```
bilingualdata1<-melt(select(d1_bilingual, HLA_CH:UAI_CH),
                     variable.name="Construct",value.name="Rating")
# Transform data into the long format
bilingualdata1$Culture <- 0
# Add a column 'Culture', '0' is the group of Chinese speakers
bilingualdata1 <- cbind(AgentID = 0, bilingualdata1)
bilingualdata1 <- cbind(agentName = 'ASIMO', bilingualdata1)

bilingualdata1$ConstructID[bilingualdata1$Construct=="HLA_CH"]<-1
bilingualdata1$ConstructID[bilingualdata1$Construct=="HLB_CH"]<-2
bilingualdata1$ConstructID[bilingualdata1$Construct=="NA_CH"]<-3
bilingualdata1$ConstructID[bilingualdata1$Construct=="NB_CH"]<-4
```

```
bilingualdata1$ConstructID[bilingualdata1$Construct=="AAS_CH"]<-5
bilingualdata1$ConstructID[bilingualdata1$Construct=="AU_CH"]<-6
bilingualdata1$ConstructID[bilingualdata1$Construct=="PF_CH"]<-7
bilingualdata1$ConstructID[bilingualdata1$Construct=="AL_CH"]<-8
bilingualdata1$ConstructID[bilingualdata1$Construct=="AS_CH"]<-9
bilingualdata1$ConstructID[bilingualdata1$Construct=="APP_CH"]<-10
bilingualdata1$ConstructID[bilingualdata1$Construct=="UAA_CH"]<-11
bilingualdata1$ConstructID[bilingualdata1$Construct=="AE_CH"]<-12
bilingualdata1$ConstructID[bilingualdata1$Construct=="UE_CH"]<-13
bilingualdata1$ConstructID[bilingualdata1$Construct=="UT_CH"]<-14
bilingualdata1$ConstructID[bilingualdata1$Construct=="UAL_CH"]<-15
bilingualdata1$ConstructID[bilingualdata1$Construct=="AA_CH"]<-16
bilingualdata1$ConstructID[bilingualdata1$Construct=="AC_CH"]<-17
bilingualdata1$ConstructID[bilingualdata1$Construct=="AI_CH"]<-18
bilingualdata1$ConstructID[bilingualdata1$Construct=="AT_CH"]<-19
bilingualdata1$ConstructID[bilingualdata1$Construct=="SP_CH"]<-20
bilingualdata1$ConstructID[bilingualdata1$Construct=="IIS_CH"]<-21
bilingualdata1$ConstructID[bilingualdata1$Construct=="AEI_CH"]<-22
bilingualdata1$ConstructID[bilingualdata1$Construct=="UEP_CH"]<-23
bilingualdata1$ConstructID[bilingualdata1$Construct=="UAI_CH"]<-24
# Label ConstructID for constructs/dimensions
```

3.2 Transforming Dutch data

First omit all unnecessary columns

```
dd2 <- data.frame(select(dd2, AC_English_1:Q_ITEMS_438.0))
dd3 <- data.frame(select(dd3, AC_English_1:Q42_442))
```

As some of the data contains the labels as well ("Disagree -3" instead of "-3"), we deleted all text from the cells

```
dd2 <- as.data.frame(lapply(dd2[1:102], function(y)
  as.numeric(gsub('[a-zA-Z]', '', y))))
dd3 <- as.data.frame(lapply(dd3[1:106], function(y)
  as.numeric(gsub('[a-zA-Z]', '', y))))
```

The naming of the dutch data is also hard to read, so in the same way as the Chinese data, we renamed them to the English ASA questionnaire tags. The datafile also contains data from the Dutch translated questionnaire, but we only need the data from English questionnaire filled in by Dutch speakers.

```
colnames(dd2)[8:51] <- c("HLA1", "HLA2", "HLA3", "HLA4", "HLB1", "HLB2", "HLB3",
                         "HLB4", "HLB5", "NA1", "NA2", "NA3", "NA4", "NA5",
                         "NB1", "NB2", "NB3", "AAS1", "AAS2", "AAS3", "AU1",
                         "AU2", "AU3", "PF1", "PF2", "PF3", "AL1", "AL2",
                         "R_AL3", "AL4", "AL5", "AS1", "AS2", "AS3", "APP1",
                         "R_APP2", "APP3", "UAA1", "UAA2", "R_UAA3", "R_AE1",
                         "AE2", "AE3", "R_AE4")

colnames(dd3)[8:53] <- c("UE1", "UE2", "UE3", "UT1", "UT2", "UT3", "UAL1",
                         "UAL2", "UAL3", "UAL4", "UAL5", "UAL6", "AA1", "AA2",
                         "AA3", "R_AC1", "R_AC2", "R_AC3", "R_AC4", "AI1",
                         "AI2", "R_AI3", "AI4", "AT1", "AT2", "R_AT3", "SP1",
                         "SP2", "SP3", "IIS1", "IIS2", "IIS3", "IIS4", "AEI1",
                         "AEI2", "R_AEI3", "AEI4", "R_AEI5", "UEP1", "UEP2",
                         "UEP3", "UEP4", "UAI1", "UAI2", "UAI3", "UAI4")
```

Reverse all columns with the name starting with "R_", because questions like 'I don't like [the agent]' and 'I like [the agent]' will have opposite values so the values of questions like 'i don't like [the agent]' are reversed.

```
for (i in grep("R_",colnames(dd2))){
# Find column number of reversing-scoring items and translations
  dd2[[i]][] <- dd2[[i]][]*(-1)
  # Reverse scores of reverse-scoring items and translations
}

for (i in grep("R_",colnames(dd3))){
# Find column number of reversing-scoring items and translations
  dd3[[i]][] <- dd3[[i]][]*(-1)
  # Reverse scores of reverse-scoring items and translations
}
```

In the same way as before, select all items that belong to the same construct and calculate the average scores of each participant of each construct.

```
d2_bilingual <-dd2
HLA_DU<-data.frame(select(d2_bilingual,HLA1:HLA4))
HLB_DU<-data.frame(select(d2_bilingual,HLB1:HLB5))
NA_DU<-data.frame(select(d2_bilingual,NA1:NA5))
NB_DU<-data.frame(select(d2_bilingual,NB1:NB3))
```

```
AAS_DU<-data.frame(select(d2_bilingual,AAS1:AAS3))
AU_DU<-data.frame(select(d2_bilingual,AU1:AU2,AU3))
PF_DU<-data.frame(select(d2_bilingual,PF1:PF3))
AL_DU<-data.frame(select(d2_bilingual,AL1:AL5))
AS_DU<-data.frame(select(d2_bilingual,AS1:AS3))
APP_DU<-data.frame(select(d2_bilingual,APP1:APP3))
UAA_DU<-data.frame(select(d2_bilingual,UAA1:R_UAA3))
AE_DU<-data.frame(select(d2_bilingual,R_AE1:R_AE4))
# Select English item scores of each construct/dimension for Construct 1-8

d3_bilingual <-dd3
UE_DU<-data.frame(select(d3_bilingual,UE1:UE3))
UT_DU<-data.frame(select(d3_bilingual,UT1:UT3))
UAL_DU<-data.frame(select(d3_bilingual,UAL1:UAL6))
AA_DU<-data.frame(select(d3_bilingual,AA1:AA3))
AC_DU<-data.frame(select(d3_bilingual,R_AC1:R_AC4))
AI_DU<-data.frame(select(d3_bilingual,AI1:AI4))
AT_DU<-data.frame(select(d3_bilingual,AT1:R_AT3))
SP_DU<-data.frame(select(d3_bilingual,SP1:SP3))
IIS_DU<-data.frame(select(d3_bilingual,IIS1:IIS4))
AEI_DU<-data.frame(select(d3_bilingual,AEI1:R_AEI5))
UEP_DU<-data.frame(select(d3_bilingual,UEP1:UEP4))
UAI_DU<-data.frame(select(d3_bilingual,UAI1:UAI4))
# Select English item scores of each construct/dimension for Construct 9-19

d2_bilingual$HLA_DU<-rowMeans(HLA_DU)
d2_bilingual$HLB_DU<-rowMeans(HLB_DU)
d2_bilingual$NA_DU<-rowMeans(NA_DU)
d2_bilingual$NB_DU<-rowMeans(NB_DU)
d2_bilingual$AAS_DU<-rowMeans(AAS_DU)
d2_bilingual$AU_DU<-rowMeans(AU_DU)
d2_bilingual$PF_DU<-rowMeans(PF_DU)
d2_bilingual$AL_DU<-rowMeans(AL_DU)
d2_bilingual$AS_DU<-rowMeans(AS_DU)
d2_bilingual$APP_DU<-rowMeans(APP_DU)
d2_bilingual$UAA_DU<-rowMeans(UAA_DU)
d2_bilingual$AE_DU<-rowMeans(AE_DU)
# Calculate English mean of each construct/dimension per participant
```

```
d3_bilingual$UE_DU<-rowMeans(UE_DU)
d3_bilingual$UT_DU<-rowMeans(UT_DU)
d3_bilingual$UAL_DU<-rowMeans(UAL_DU)
d3_bilingual$AA_DU<-rowMeans(AA_DU)
d3_bilingual$AC_DU<-rowMeans(AC_DU)
d3_bilingual$AI_DU<-rowMeans(AI_DU)
d3_bilingual$AT_DU<-rowMeans(AT_DU)
d3_bilingual$SP_DU<-rowMeans(SP_DU)
d3_bilingual$IIS_DU<-rowMeans(IIS_DU)
d3_bilingual$AEI_DU<-rowMeans(AEI_DU)
d3_bilingual$UEP_DU<-rowMeans(UEP_DU)
d3_bilingual$UAI_DU<-rowMeans(UAI_DU)
# Calculate English mean of each construct/dimension per participant
```

We added the culture tag 1 for the Dutch data and the ConstructID for each construct and change the data to a long format for the first 12 constructs/dimensions

```
bilingualdata2<-melt(select(d2_bilingual, HLA_DU:AE_DU),
                     variable.name="Construct",value.name="Rating")
# Transform data into the long format
bilingualdata2$Culture <- 1
# Add a column 'Culture', '1' is participant group of Dutch speakers
bilingualdata2 <- cbind(AgentID = 0, bilingualdata2)
bilingualdata2 <- cbind(agentName = 'ASIMO', bilingualdata2)

bilingualdata2$ConstructID[bilingualdata2$Construct=="HLA_DU"]<-1
bilingualdata2$ConstructID[bilingualdata2$Construct=="HLB_DU"]<-2
bilingualdata2$ConstructID[bilingualdata2$Construct=="NA_DU"]<-3
bilingualdata2$ConstructID[bilingualdata2$Construct=="NB_DU"]<-4
bilingualdata2$ConstructID[bilingualdata2$Construct=="AAS_DU"]<-5
bilingualdata2$ConstructID[bilingualdata2$Construct=="AU_DU"]<-6
bilingualdata2$ConstructID[bilingualdata2$Construct=="PF_DU"]<-7
bilingualdata2$ConstructID[bilingualdata2$Construct=="AL_DU"]<-8
bilingualdata2$ConstructID[bilingualdata2$Construct=="AS_DU"]<-9
bilingualdata2$ConstructID[bilingualdata2$Construct=="APP_DU"]<-10
bilingualdata2$ConstructID[bilingualdata2$Construct=="UAA_DU"]<-11
bilingualdata2$ConstructID[bilingualdata2$Construct=="AE_DU"]<-12
# Label ConstructID for English constructs/dimensions
```

in the same fashion as the first 12 constructs/dimensions, we perform the same data manipulation on

the last 12 constructs/dimensions

```
bilingualdata3<-melt(select(d3_bilingual, UE_DU:UAI_DU),
                     variable.name="Construct",value.name="Rating")
# Transform data into the long format
bilingualdata3$Culture <- 1
# Add a column 'Culture', '1' is participant group of Dutch speakers
bilingualdata3 <- cbind(AgentID = 0, bilingualdata3)
bilingualdata3 <- cbind(agentName = 'ASIMO', bilingualdata3)

bilingualdata3$ConstructID[bilingualdata3$Construct=="UE_DU"]<-13
bilingualdata3$ConstructID[bilingualdata3$Construct=="UT_DU"]<-14
bilingualdata3$ConstructID[bilingualdata3$Construct=="UAL_DU"]<-15
bilingualdata3$ConstructID[bilingualdata3$Construct=="AA_DU"]<-16
bilingualdata3$ConstructID[bilingualdata3$Construct=="AC_DU"]<-17
bilingualdata3$ConstructID[bilingualdata3$Construct=="AI_DU"]<-18
bilingualdata3$ConstructID[bilingualdata3$Construct=="AT_DU"]<-19
bilingualdata3$ConstructID[bilingualdata3$Construct=="SP_DU"]<-20
bilingualdata3$ConstructID[bilingualdata3$Construct=="IIS_DU"]<-21
bilingualdata3$ConstructID[bilingualdata3$Construct=="AEI_DU"]<-22
bilingualdata3$ConstructID[bilingualdata3$Construct=="UEP_DU"]<-23
bilingualdata3$ConstructID[bilingualdata3$Construct=="UAI_DU"]<-24
# Label ConstructID for English constructs/dimensions
```

In the end we combine all 3 datasets into 1 big dataset

```
data_culture <- rbind(bilingualdata1, bilingualdata2, bilingualdata3)
data_culture <- na.omit(data_culture)
# Combine data 'bilingualdata1', 'bilingualdata2' and 'mixdata'
data_culture <- select(data_culture, -Construct)

write_sav(data_culture, "data_cultures.sav",
          compress = c("byte", "none", "zsav"))
```