German translation of the Artificial-Social-Agent questionnaire instrument for evaluating human-agent interaction

Underlying Analyses - second translation round

Boleslav Khodakov

31 May, 2023

Contents

1 Introduction

This document presents statistical analyses of correlation and variation between English and German ASA questionnaires for the item level. For the second translation round, the translations of 35 English items are evaluated. There are a total of 78 translations, as all items have multiple alternative translations. **Please note: The descriptions provided in this document relate to the dataset containing all participants' data. However, there is also a subset of 29 participants who recommended using their data. This subset is evaluated in the same way until (including) individual ICC values are calculated. After that, we only continue with the complete (30 participant) data.**

We use the following packages:

```
library(nlme)     # Run multilevel linear models
library(car)      # Package linear regression
library(rethinking) # Run ulam
library(haven) # Use read_sav fuction
```

```r
library(dplyr) # Use select 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 file

The input data used in the analysis were transformed from the raw data file 'Final_ASA_German_Round_2_anonym.s detailed transformation from raw data to the input data file was explained in the markdown file 'Transformation from raw data to the input data files'.

2.1 File transformed_data_round_2.sav

Human-ASA interaction evaluation data were collected from 30 bilingual participants with German mother tongue who are native German and fluent English speakers. Bilingual participants rated human-ASA interaction on 35 English items and corresponding German translations plus 14 attention control questions. All participants' evaluation data were included as they failed no attention control questions. We removed irrelevant data, e.g., attention control questions, just retaining scores of English items and corresponding German translations, also with 'AgentID'. We did not yet invert reverse-scoring questionnaire items and their English translations. The steps above were conducted and explained in the markdown file 'Transformation from raw data to the input data files', resulting in a single data file 'transformed_data_round_2.sav'. Up to this step, rating scores of 35 English items and corresponding German translations were ready for further analysis.

```r
# Load data for all participants
data01 <- data.frame(read_sav("transformed_data_round_2.sav"))
# Load data for "recommended" participants
data02 <- data.frame(read_sav("transformed_data_round_2_recommended.sav"))
# Select item scores for English and German translation scores
d1 <- select(data01, Q_E_HLA4:Q_DE_UAI4_3)
d2 <- select(data02, Q_E_HLA4:Q_DE_UAI4_3)
```

3 Analyses results

3.1 Correlation between English and German ASA Questionnaire

We combined the scores of 35 items as well as their corresponding translations in dataframe 'd1'. Then we calculated ICC values for the 35 items. The multilevel model that we fit on the data set is a random intercept model. This model includes a fixed intercept (~1) and participant as a random intercept, indicated by random = ~1|id. Here, 'id' indicates the participant code for 30 bilingual participants whose scores were used to calculate ICC values.

3.1.1 ICC values for 78 items

We calculated ICC values for the 35 items. Items were duplicated leading to a total of 78 (so each of the multiple translations had one match). The multilevel model that we fit on the data set is a random intercept model. This model includes a fixed intercept (~1) and participant as a random intercept, indicated by random = ~1|id. Here, 'id' indicates the participant code for 30 bilingual participants whose scores were used to calculate ICC values. We calculated ICC as: $\rho_I = \frac{\tau^2}{\tau^2+\sigma^2}$ whereby τ^2 is the variance between participants, and σ^2 is the variance within the score of individual (Finch, Bolin, and Kelley 2019). For the ICC calculation we defined the *getICC* function.

```
getICC <-function(model)
# Function for ICC value calculation using multilevel linear model
{
  vc.model <- VarCorr(model)
  # Estimated variances and correlations between the random-effects terms
  sigma_var <-as.numeric(vc.model[2,1])
  # Variance within the groups
  tau_var <- as.numeric(vc.model[1,1])
  # Variance between the groups
  icc <- tau_var/(tau_var + sigma_var)
  # Calculate ICC value
  return(icc)
}
```

Next, we defined a function to run a multilevel model and obtain the associated ICC value for that model. As input, this function accepts the scores in both languages and the participant ID number. Before the model can be fitted this input data is transformed into a long format. The function returns ICC in value.

```
getLME <-function(s_1,s_2)
# Function for a linear mixed-effects model
{
  id<-rownames(s_2)
  # Row names that represent the ID number of each participant
  score_German<- data.frame(id, s_1, language= 1)
  # Transform German scores from wide format to long format and label as 1
  Score_English<- data.frame(id, s_2, language= 2)
  # Transform English scores from wide format to long format and label as 2
```

```
  Score_total <- rbind(score_German, Score_English)
  # Combine German and English scores in the long format
  m0 <- lme(score ~ 1, data = Score_total, random = ~1|id, method = "ML")
  # Linear mixed-effects model with a fixed intercept and
  # a random intercept of participant's ID number
  return(getICC(m0))
}
```

With the *getLME* function defined, the next step is to use this function to calculate the ICC value for each of the 35 (78, counting duplicated columns for multiple translations) ASA questionnaire items, and in addition, calculate the grand mean of these 78 ICC values. When going to the list of ASAQ items, we use the fact that in the data frame the first 78 columns (incl. duplicates) present the results of the English ASAQ version and the last 78 (unique) columns present the results of the German ASAQ version. **Please note: There is no simple way of displaying only the best (highest-ICC) German items for English items with multiple translations. Below, you will find all translations.**

```
# Calculate individual ICC values
calculate_item_ICC_values <- function(data,
                                      icc_output_caption, all_participants){

  l_ICC <- data.frame(ItemID = double(), Item = character(), icc = double())

  #Initialize output file for low-ICC combinations
  write(icc_output_caption,file="ICC_output.txt",append=TRUE)



  # Numbers of columns in d_total
  German_column_offset <- ncol(data) /2



  # The value of n is equal to the number of columns divided by 2.
  n <- 78
  for (i in 1:n)
  # Go step by step to 78 items
  # whereby i is the ASA questionnaire item number
  {

    # Select scores of German version of ASAQ item i
    score_German <- data.frame(score=data[,i + German_column_offset])
```

```
    # Select scores of English version of ASAQ items i
    score_English <- data.frame(score=data[,i])



    # Calculated ICC and add it to the list of ICC values,
    # with ID number of the ASA questionnaire item
    l_ICC <- rbind(l_ICC, data.frame (i, icc = getLME(score_German, score_English)))

    # Get the current ICC value from l_ICC (which is a table)
    real_ICC <- round(l_ICC[i,2], digits=4)

    # For values which are under the threshold of 'Good' (0.6)
    if(0.6 > real_ICC){

      # Create an entry in a text file, to have a list of dissatisfactory translations
      output1 <- paste("l_ICC", real_ICC, sep=" ")
      output1 <- paste(output1, colnames(data[i]), sep=" ")
      output1 <- paste(output1, colnames(data[i + German_column_offset]), sep=" ")

      # Append a text file (of the user's choice) with the entry
      write(output1,file="ICC_output.txt",append=TRUE)
    }

  }

  l_ICC$Item = colnames(select(data,Q_E_HLA4:Q_E_UAI4_3)) # Add name code for each item

  return(l_ICC)
}
```

We also calculate item-level ICC values for only the recommended data rows.

```
# Calculate individual ICC values for all participants
l_ICC <- calculate_item_ICC_values(d1, "low-ICCs for all participants", TRUE)

pander(l_ICC, caption = "All participants - ICC values for 78 items")
```

Table 1: All participants - ICC values for 78 items

i	icc	Item
1	0.5442	Q_E_HLA4
2	0.5497	Q_E_HLA4_2
3	0.5344	Q_E_NA1
4	0.3238	Q_E_NA1_2
5	0.7719	Q_E_NA2
6	0.6369	Q_E_NA2_2
7	0.7483	Q_E_NA4
8	0.851	Q_E_NA4_2
9	0.8646	Q_E_NA5
10	0.8517	Q_E_NA5_2
11	0.6716	Q_E_NB1
12	0.4614	Q_E_NB1_2
13	0.7709	Q_E_AAS1
14	0.7799	Q_E_AAS1_2
15	0.6821	Q_E_APP1
16	0.3998	Q_E_APP1_2
17	0.6542	Q_E_APP2
18	0.7063	Q_E_APP2_2
19	0.7084	Q_E_UAA3
20	0.3611	Q_E_UAA3_2
21	0.7173	Q_E_AE4
22	0.693	Q_E_AE4_2
23	0.3824	Q_E_AE4_3
24	0.6657	Q_E_UE1
25	0.4845	Q_E_UE1_2
26	0.2689	Q_E_UE2
27	0.5229	Q_E_UE2_2
28	0.6619	Q_E_UE3
29	0.3514	Q_E_UE3_2
30	0.6546	Q_E_UAL2
31	0.7124	Q_E_UAL2_2
32	0.5755	Q_E_UAL3
33	0.591	Q_E_UAL3_2
34	0.4388	Q_E_UAL5
35	0.2316	Q_E_UAL5_2
36	0.4545	Q_E_AA1

i	icc	Item
37	0.601	Q_E_AA1_2
38	0.5353	Q_E_AA1_3
39	0.7068	Q_E_AI1
40	0.6584	Q_E_AI1_2
41	0.6197	Q_E_AI1_3
42	0.6039	Q_E_AI3
43	0.7541	Q_E_AI3_2
44	0.717	Q_E_AI4
45	0.7422	Q_E_AI4_2
46	0.7174	Q_E_AI4_3
47	0.8266	Q_E_AT2
48	0.6479	Q_E_AT2_2
49	0.5931	Q_E_AT3
50	0.7619	Q_E_AT3_2
51	0.7626	Q_E_SP2
52	0.6153	Q_E_SP2_2
53	0.7421	Q_E_SP3
54	0.6619	Q_E_SP3_2
55	0.7241	Q_E_SP3_3
56	0.4633	Q_E_IIS3
57	0.5008	Q_E_IIS3_2
58	0.5183	Q_E_IIS4
59	0.5762	Q_E_IIS4_2
60	0.604	Q_E_AEI1
61	0.6	Q_E_AEI1_2
62	0.4393	Q_E_AEI3
63	0.2269	Q_E_AEI3_2
64	0.5167	Q_E_AEI4
65	0.6629	Q_E_AEI4_2
66	0.3553	Q_E_AEI5
67	0.1048	Q_E_AEI5_2
68	0.1297	Q_E_AEI5_3
69	0.6869	Q_E_UAI1
70	0.4625	Q_E_UAI1_2
71	0.5986	Q_E_UAI1_3
72	0.5677	Q_E_UAI2
73	0.3703	Q_E_UAI2_2
74	0.7774	Q_E_UAI3

i	icc	Item
75	0.6948	Q_E_UAI3_2
76	0.6333	Q_E_UAI4
77	0.5638	Q_E_UAI4_2
78	0.654	Q_E_UAI4_3

```
Variable <- c("Grand_mean","SD","Minimum","Maximum")
# Define the names of the statistics
Value <- c(round(mean(l_ICC$icc),digits=4),round(sd(l_ICC$icc),digits=4),
           round(min(l_ICC$icc),digits=4),round(max(l_ICC$icc),digits=4))
# Calculate the grand mean, standard deviation,
# minimum and maximum values of ICC values of 78 items
description <- cbind(Variable, Value) # Descriptive statistics of ICC values of 78 items

# Print results
pander(description, caption = paste("All participants - Descriptive statistics",
                        " of ICC values of 35 (78) items"))
```

Table 2: All participants - Descriptive statistics of ICC values of 35 (78) items

Variable	Value
Grand_mean	0.5895
SD	0.1656
Minimum	0.1048
Maximum	0.8646

```
# Calculate individual ICC values for "recommended" participants
l_ICC_recommended <- calculate_item_ICC_values(d2, "low-ICCs for recommended participants", FALSE)

pander(l_ICC_recommended, caption = "Recommended participants - ICC values for 78 items")
```

Table 3: Recommended participants - ICC values for 78 items

i	icc	Item
1	0.555	Q_E_HLA4
2	0.5605	Q_E_HLA4_2
3	0.5456	Q_E_NA1

i	icc	Item
4	0.3302	Q_E_NA1_2
5	0.7704	Q_E_NA2
6	0.6367	Q_E_NA2_2
7	0.7715	Q_E_NA4
8	0.8554	Q_E_NA4_2
9	0.8686	Q_E_NA5
10	0.8541	Q_E_NA5_2
11	0.6691	Q_E_NB1
12	0.4531	Q_E_NB1_2
13	0.7762	Q_E_AAS1
14	0.8263	Q_E_AAS1_2
15	0.6816	Q_E_APP1
16	0.4025	Q_E_APP1_2
17	0.6959	Q_E_APP2
18	0.7516	Q_E_APP2_2
19	0.7125	Q_E_UAA3
20	0.3588	Q_E_UAA3_2
21	0.7218	Q_E_AE4
22	0.6974	Q_E_AE4_2
23	0.3842	Q_E_AE4_3
24	0.6618	Q_E_UE1
25	0.4752	Q_E_UE1_2
26	0.1754	Q_E_UE2
27	0.5356	Q_E_UE2_2
28	0.64	Q_E_UE3
29	0.2872	Q_E_UE3_2
30	0.6413	Q_E_UAL2
31	0.7498	Q_E_UAL2_2
32	0.5725	Q_E_UAL3
33	0.596	Q_E_UAL3_2
34	0.4775	Q_E_UAL5
35	0.2069	Q_E_UAL5_2
36	0.4563	Q_E_AA1
37	0.604	Q_E_AA1_2
38	0.5378	Q_E_AA1_3
39	0.7036	Q_E_AI1
40	0.6539	Q_E_AI1_2
41	0.6225	Q_E_AI1_3

i	icc	Item
42	0.603	Q_E_AI3
43	0.7535	Q_E_AI3_2
44	0.7148	Q_E_AI4
45	0.8142	Q_E_AI4_2
46	0.7079	Q_E_AI4_3
47	0.8271	Q_E_AT2
48	0.6453	Q_E_AT2_2
49	0.6606	Q_E_AT3
50	0.7555	Q_E_AT3_2
51	0.7694	Q_E_SP2
52	0.6169	Q_E_SP2_2
53	0.7691	Q_E_SP3
54	0.6693	Q_E_SP3_2
55	0.7919	Q_E_SP3_3
56	0.4624	Q_E_IIS3
57	0.5038	Q_E_IIS3_2
58	0.5176	Q_E_IIS4
59	0.5755	Q_E_IIS4_2
60	0.5965	Q_E_AEI1
61	0.594	Q_E_AEI1_2
62	0.5286	Q_E_AEI3
63	0.2167	Q_E_AEI3_2
64	0.523	Q_E_AEI4
65	0.6592	Q_E_AEI4_2
66	0.5991	Q_E_AEI5
67	0.109	Q_E_AEI5_2
68	0.1186	Q_E_AEI5_3
69	0.6897	Q_E_UAI1
70	0.4624	Q_E_UAI1_2
71	0.6111	Q_E_UAI1_3
72	0.5663	Q_E_UAI2
73	0.3726	Q_E_UAI2_2
74	0.7819	Q_E_UAI3
75	0.6947	Q_E_UAI3_2
76	0.6476	Q_E_UAI4
77	0.5592	Q_E_UAI4_2
78	0.6506	Q_E_UAI4_3

```
Variable <- c("Grand_mean","SD","Minimum","Maximum")
# Define the names of the statistics
Value <- c(round(mean(l_ICC_recommended$icc),digits=4),round(sd(l_ICC_recommended$icc),digits=4),
           round(min(l_ICC_recommended$icc),digits=4),round(max(l_ICC_recommended$icc),digits=4))
# Calculate the grand mean, standard deviation,
# minimum and maximum values of ICC values of 78 items
description <- cbind(Variable, Value) # Descriptive statistics of ICC values of 78 items

# Print results
pander(description, caption = paste("Recommended participants - Descriptive statistics",
                        " of ICC values of 35 (78) items"))
```

Table 4: Recommended participants - Descriptive statistics of ICC values of 35 (78) items

Variable	Value
Grand_mean	0.5976
SD	0.172
Minimum	0.109
Maximum	0.8686

For the assessment of the correlation between the English and German ASA Questionnaire, we followed Cicchetti's classification of ICC categories (Cicchetti 1994). Then we get the categories of ICC classifications and number of ICC values in classification category. **Please note: There is no simple way of displaing only the best (highest-ICC) German items for English items with multiple translations. Below, you will find the classification of all translations.**

```
Classification <- c("Excellent","Good","Fair","Poor")
ICC_Range <- c("0.75-1.00","0.60-0.74","0.40-0.59","0-0.39")
# Categories of ICC classifications by Cicchetti (1994)
n_item <- length(l_ICC$icc) # Number of ICC values
round_ICC <- round(l_ICC$icc, digits=4) # Round ICC values
Number <- c(length(l_ICC[which(round_ICC>=0.75&round_ICC<=1),]$icc),
            length(l_ICC[which(round_ICC>=0.60&round_ICC<=0.74),]$icc),
            length(l_ICC[which(round_ICC>=0.40&round_ICC<=0.59),]$icc),
            length(l_ICC[which(round_ICC>=0.00&round_ICC<=0.39),]$icc))
# Calculate number of ICC values in classification category
Percentage <- c(round(Number[1]/n_item,digits=4)*100, round(Number[2]/n_item,digits=4)*100,
                round(Number[3]/n_item,digits=4)*100, round(Number[4]/n_item,digits=4)*100)
# Calculate percentage of ICC values in classification category
```

```r
ICC_category <- cbind(Classification,ICC_Range,Number,Percentage)

# Print results
pander(ICC_category, caption = "Categories of ICC classifications and
       number of ICC values in classification category for 78 items")
```

Table 5: Categories of ICC classifications and number of ICC values in classification category for 78 items

Classification	ICC_Range	Number	Percentage
Excellent	0.75-1.00	11	14.1
Good	0.60-0.74	30	38.46
Fair	0.40-0.59	19	24.36
Poor	0-0.39	11	14.1

References

Cicchetti, Domenic V. 1994. "Guidelines, Criteria, and Rules of Thumb for Evaluating Normed and Standardized Assessment Instruments in Psychology." *Psychological Assessment* 6 (4): 284. https://doi.org/10.1037/1040-3590.6.4.284.

Finch, W Holmes, Jocelyn E Bolin, and Ken Kelley. 2019. *Multilevel Modeling Using r.* Crc Press.