

German and Dutch Translations of the Artificial-Social-Agent Questionnaire Instrument for Evaluating Human-Agent Interactions

Transformation from German raw data to the input files - first half

Nele Albers, Boleslav A. Khodakov

18 June, 2024

Contents

Introduction	1
Load packages	1
Read raw data	2
Merge item versions	2
Mark and reverse reverse-scored items	3
Save final data file	4

Introduction

In this document, we transform the anonymized German data from the first half of the questionnaire into an input data file for our analysis. Because of privacy reasons, the non-anonymized raw data is not publicly available.

There are two raw data files to be transformed. They are divided like the questionnaire - split into two parts. Two groups of participants answered either questionnaire. In the first group, which this document is about, human-ASA interaction evaluation data of the first 44 items (Construct 1-8: the first 12 constructs/dimensions) were collected from bilingual participants with German as their primary language, and English as their fluent language. Bilingual participants rated human-ASA interaction on 44 English items and corresponding German translations. In addition, they answered 14 attention control questions.

Required files: Data/Final_ASA_German_Summative_First_Half_final_2023_07_15_anonym.csv

Created files: summative_first_half_transformed_german.sav

Authored by Nele Albers, Andrea Bönsch, Jonathan Ehret, Boleslav A. Khodakov, and Willem-Paul Brinkman.

Load packages

Let's load the packages that we need.

```
library(dplyr)      # Use select function
library(formatR)    # For formatting
library(haven)      # Use read_sav function
```

Read raw data

We read the anonymized raw data file consisting of the first 44 English ASA item scores and corresponding German translation scores. The raw scores are from a 7-point scale, ranging from -3 (disagree) to 3 (agree).

```
df <- read.csv("Data/Final_ASA_German_Summative_First_Half_final_2023_07_15_anonym.csv")
```

Merge item versions

Because of different item formulations depending on the number of agents, agent gender, and human gender of the 14 rated human-agent interaction videos, there are multiple versions of the same questionnaire (2 English versions, 7 German versions). While all versions of the questionnaire were present in one survey, their columns are currently not merged. The code below allows to merge the separate versions of a questionnaire.

```
merge_data <- function(data, multiplier = 0) {
  start <- 1 # this is the column of the first English item (the attention check)
  end <- start + 50 # this is the last column of the English items
  # Create a new list (this will be the new columns with
  # combined data)
  new_columns <- list()

  # Go over all columns
  for (i in seq(start, end)) {
    # Get the column name without suffix
    col1 <- colnames(data)[i]
    # Get column index for next suffixed column of same
    # name
    col2_index <- i + 51
    # Get its name
    col2 <- colnames(data)[col2_index]

    # Use the name of the unsuffixed column
    new_col <- colnames(data)[i]

    # Create a new column by merging the values of col1
    # and col2 by checking if they are NA The name of
    # col1, the unsuffixed column, will be the name of
    # the merged column

    new_columns[[new_col]] <- ifelse(is.na(data[[col1]]),
                                     data[[col2]], data[[col1]])
  }

  # Make list of columns into a data frame
  merged_data <- data.frame(new_columns)
  # Remove the columns that were merged, and replace them
  # with combined columns Keep any remaining columns
  # (appended), to repeat this operation There are 102
  # columns
  combined_data <- cbind(merged_data, data[-1:-102])
}
```

```

    return(combined_data)
}

```

Merge the two versions of the English questionnaire into one.

```

# Select only questions, attention checks
df_E <- data.frame(select(df, AttentionCheck_E_1:"Q_E_AE4",
                          "AttentionCheck_E_1.1":"Q_E_AE4.1"))

# Merge suffixed columns into one
df_E <- merge_data(df_E)

```

Merge the versions of the German questionnaire into one.

```

# Select only German questions, attention checks
df_DE <- select(df, AttentionCheck_DE_1:Q_DE_AE4, AttentionCheck_DE_1.1:Q_DE_AE4.1,
               AttentionCheck_DE_1.2:Q_DE_AE4.2, AttentionCheck_DE_1.3:Q_DE_AE4.3,
               AttentionCheck_DE_1.4:Q_DE_AE4.4, AttentionCheck_DE_1.5:Q_DE_AE4.5,
               AttentionCheck_DE_1.6:Q_DE_AE4.6)

# Merge all variants of German (suffixed) columns into one
df0 <- merge_data(df_DE)

df1 <- merge_data(df0)

df2 <- merge_data(df1)

df3 <- merge_data(df2)

df4 <- merge_data(df3)

df5 <- merge_data(df4)

# Rename for better legibility
df_DE <- df5

```

Combine the unified English and German items.

```

# Combine English and German questions + attention checks
d_ASA_1 <- cbind(df_E, df_DE)
# Make entries numeric
d_ASA_1 <- d_ASA_1 %>% mutate_if(is.character, as.numeric)

```

Mark and reverse reverse-scored items

Here, we now add the prefix R to items which have the sign [R] in the ASA questionnaire (i.e., are reverse-scored). Then, the scores of these items are reversed.

```

# Select only question items from data frame (incl.
# attention checks)
dd1 <- data.frame(cbind(select(d_ASA_1, AttentionCheck_E_1:Q_DE_AE4),
                          select(df, agentName)))

# Add R prefix to reversely scored items
colnames(dd1)[colnames(dd1) == "Q_E_AL3"] = "Q_E_R_AL3"

```

```

colnames(dd1)[colnames(dd1) == "Q_E_APP2"] = "Q_E_R_APP2"
colnames(dd1)[colnames(dd1) == "Q_E_UAA3"] = "Q_E_R_UAA3"
colnames(dd1)[colnames(dd1) == "Q_E_AE1"] = "Q_E_R_AE1"
colnames(dd1)[colnames(dd1) == "Q_E_AE4"] = "Q_E_R_AE4"

colnames(dd1)[colnames(dd1) == "Q_DE_AL3"] = "Q_DE_R_AL3"
colnames(dd1)[colnames(dd1) == "Q_DE_APP2"] = "Q_DE_R_APP2"
colnames(dd1)[colnames(dd1) == "Q_DE_UAA3"] = "Q_DE_R_UAA3"
colnames(dd1)[colnames(dd1) == "Q_DE_AE1"] = "Q_DE_R_AE1"
colnames(dd1)[colnames(dd1) == "Q_DE_AE4"] = "Q_DE_R_AE4"

# Find column number of reverse-scoring items and
# translations
for (i in grep("R", colnames(dd1))) {
  # Reverse scores of reverse-scoring items and
  # translations
  dd1[[i]][,] <- dd1[[i]][,] * (-1)
}

# Add a column 'AgentID' to facilitate analysis for
# comparison between different cultural backgrounds in the
# main markdown file No specific meaning for 14 numbers,
# just a code for each agent
dd1$AgentID <- 0
# Label AgentID for 14 ASAs
dd1$AgentID[dd1$agentName == "iCAT"] <- 1
dd1$AgentID[dd1$agentName == "DEEPBLUE"] <- 2
dd1$AgentID[dd1$agentName == "AMY"] <- 3
dd1$AgentID[dd1$agentName == "FURBY"] <- 4
dd1$AgentID[dd1$agentName == "POPPIE"] <- 5
dd1$AgentID[dd1$agentName == "SIRI"] <- 6
dd1$AgentID[dd1$agentName == "HAL 9000"] <- 7
dd1$AgentID[dd1$agentName == "SIM SENSEI"] <- 8
dd1$AgentID[dd1$agentName == "CHAPPIE"] <- 9
dd1$AgentID[dd1$agentName == "AIBO"] <- 10
dd1$AgentID[dd1$agentName == "SARAH"] <- 11
dd1$AgentID[dd1$agentName == "NAO"] <- 12
dd1$AgentID[dd1$agentName == "MARCUS"] <- 13
dd1$AgentID[dd1$agentName == "DOG"] <- 14
attr(dd1$agentName, "label") <- c("iCAT, DEEPBLUE, AMY, FURBY, POPPIE, SIRI,
HAL 9000, SIM SENSEI, CHAPPIE, AIBO, SARAH, NAO, MARCUS, DOG")
# Add label to 'AgentID'
attr(dd1$AgentID, "label") <- c("1=iCAT, 2=DEEPBLUE, 3=AMY, 4=FURBY, 5=POPPIE, 6=SIRI,
7=HAL 9000, 8=SIM SENSEI, 9=CHAPPIE, 10=AIBO, 11=SARAH, 12=NAO, 13=MARCUS, 14=DOG")

```

Save final data file

And we save the resulting data file.

```

write_sav(dd1, "summative_first_half_transformed_german.sav",
          compress = c("byte", "none", "zsav"))

```