

Experiment

Adarsh Denga

2025-02-26

Introduction

This document gives an overview of the analysis of the measures used for evaluating the integration of Large Language Models into a rule-based chatbot counsellor training system for child helplines. The document includes an analysis of:

- Paired-Samples T-Test: Analysis of the Believability, Engagement, Attitude and Overall Experience. The calculation and questionnaire for these constructs are available in the thesis PDF in Chapter 4 and Appendix E respectively.
- Cohen's Kappa: Analysis of the agreement between two coders for the identified themes in the thematic analysis. The calculation and list of themes are available in the thesis PDF in Chapter 4 and Appendix E respectively.
- Binomial Test: Analysis of the overall preference between the conditions. The calculation and preference statistics are given in the thesis PDF in Chapter 4.

Additionally, we also provide the code for our power analysis to determine the minimum required sample size for our experiment.

We need the following files for our analysis:

- `data_raw_llm.csv`: The raw experiment scores for our constructs for each participant from using the LLM-based system.
- `data_raw_rbs.csv`: The raw experiment scores for our constructs for each participant from using the rule-based system.

The two above files have the following columns: `PROLIFIC_PID` (an anonymised participant ID), and `Q1-Q34` (participant scores for each question in our questionnaire). We use the script in `construct_split.py` to transform the data into the data in the two next files:

- `constructs_averaged_llm.csv`: The average scores for five constructs (Overall Experience, Human-Like Behaviour, Natural Behaviour, Engagement, Attitude) for each participant from using the LLM-based system.
- `constructs_averaged_rbs.csv`: The average scores for five constructs (Overall Experience, Human-Like Behaviour, Natural Behaviour, Engagement, Attitude) for each participant from using the rule-based system.

The two above files have the following columns: `PROLIFIC_PID` (an anonymised participant ID), `OVERALL` (average score for overall experience construct), `BELIEVABILITY1` (averaged score for human-like behaviour construct), `BELIEVABILITY2` (averaged score for natural behaviour construct), `ENGAGEMENT` (averaged score for engagement construct), and `ATTITUDE` (overall score for attitude construct).

- `qual.csv`: The raw thematic feedback from our participants for the LLM and RBS systems respectively.

The above file has the following columns: `PROLIFIC_PID` (an anonymised participant ID), `CONDITION_ORDER` (a flag which determines which order the participants interacted with the system), `LLM` (their feedback from using the LLM-based system) and `RBS` (their feedback from using the rule-based system).

- `construct_split.py`: The Python script used to obtain the averaged scores for each construct from `data_raw_llm.csv` and `data_raw_rbs.csv` to `constructs_averaged_llm.csv` and `constructs_averaged_rbs.csv` respectively.

Paired-Samples T-Test

Given below is the code for our Paired Samples T-Test for our five constructs.

```
# Load the dplyr package
library(dplyr)

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##   filter, lag

## The following objects are masked from 'package:base':
##
##   intersect, setdiff, setequal, union

# Read data files for the LLM and rule-based conditions
df1 <- read.csv("constructs_averaged_llm.csv")
df2 <- read.csv("constructs_averaged_rbs.csv")

# Combine data from both files based on participant ID
merged_df <- inner_join(df1, df2, by = "PROLIFIC_PID", suffix = c("_1", "_2"))

# Perform T-Tests for the five constructs
tests <- list(
  overall = t.test(merged_df$OVERALL_1, merged_df$OVERALL_2, paired = TRUE),
  believability1 = t.test(merged_df$BELIEVABILITY1_1, merged_df$BELIEVABILITY1_2, paired = TRUE),
  believability2 = t.test(merged_df$BELIEVABILITY2_1, merged_df$BELIEVABILITY2_2, paired = TRUE),
  engagement = t.test(merged_df$ENGAGEMENT_1, merged_df$ENGAGEMENT_2, paired = TRUE),
  attitude = t.test(merged_df$ATTITUDE_1, merged_df$ATTITUDE_2, paired = TRUE)
)

# Gather relevant results
results_df <- data.frame(
  Measure = names(tests),
  t_statistic = sapply(tests, function(t) round(t$statistic, 3)),
  p_value = sapply(tests, function(t) formatC(t$p.value, format = "e", digits = 2)),
  mean_difference = sapply(tests, function(t) round(t$estimate, 3)),
  conf_low = sapply(tests, function(t) round(t$conf.int[1], 3)),
  conf_high = sapply(tests, function(t) round(t$conf.int[2], 3))
)

# Print results
print(results_df, row.names = FALSE)
```

Measure	t_statistic	p_value	mean_difference	conf_low	conf_high
overall	2.567	1.46e-02	0.478	0.100	0.855
believability1	2.098	4.29e-02	0.438	0.015	0.861
believability2	1.381	1.76e-01	0.333	-0.156	0.823
engagement	1.069	2.92e-01	0.171	-0.154	0.496
attitude	2.456	1.90e-02	0.739	0.129	1.349

Cohen's Kappa

```
# Load the irr package
library(irr)

## Loading required package: lpSolve
# Identified Coder Themes
coder1 <- c(6,3,7,9,7,6,5,6,1,10,3,8,5,3,3,8,10,10,2,3,12,7,3,7,10,3,2,1,2,1,3,6,2,2,2,1,1,
           3,7,4,2,7,6,12,11,7,3,7,6,5,7,12,7,5,12,7,1,7,6,7,6,6,3,3,12,7,7,7,12,3,2,7,6,3)
coder2 <- c(5,3,7,1,7,6,5,6,1,1,7,9,5,1,2,1,7,10,7,3,7,7,7,12,10,1,8,9,2,9,7,5,2,9,4,6,2,
           5,3,4,1,7,6,5,11,5,3,7,6,5,7,5,7,3,12,6,1,12,11,3,8,5,9,7,7,12,9,10,4,11,1,7,6,3)

# Perform Cohen's Kappa Test
kappa_value <- kappa2(data.frame(coder1, coder2))

# Print results
print(kappa_value)

## Cohen's Kappa for 2 Raters (Weights: unweighted)
##
## Subjects = 74
## Raters = 2
## Kappa = 0.315
##
## z = 8
## p-value = 1.33e-15
```

Binomial Test

Given below is the code for our binomial test.

```
# Perform binomial test
binom_results <- binom.test(26, 37, p = 0.5, alternative = "two.sided")

# Print results
print(binom_results)

##
## Exact binomial test
##
## data: 26 and 37
## number of successes = 26, number of trials = 37, p-value = 0.02007
## alternative hypothesis: true probability of success is not equal to 0.5
## 95 percent confidence interval:
## 0.5302005 0.8412746
## sample estimates:
## probability of success
## 0.7027027
```

Power Analysis

Given below is the code for our initial power analysis to determine the minimum number of participants for our paired-samples t-test with

```
# Load the pwr package
library(pwr)
```

```
# Perform power analysis for Two-Tailed Paired-Samples T-Test
# Effect Size = 0.5 (Medium)
# Target Power = 0.80
# Alpha Error = 0.05
# Type = Two-Tailed
power_analysis_result <- pwr.t.test(d = 0.5, power = 0.80, sig.level = 0.05, type = "paired", alternative = "two.sided")

# Print results
print(power_analysis_result)

##
##      Paired t test power calculation
##
##              n = 33.36713
##              d = 0.5
##      sig.level = 0.05
##              power = 0.8
##      alternative = two.sided
##
## NOTE: n is number of *pairs*
```