Relational Event Modeling (REM) Project README

2024 May

Project Directory

R Code

- data_clean.Rmd: initial data cleaning
- data_process.Rmd: specifically prepare the data for REM modeling (sender, receiver, time, etc.)
- data_viz.Rmd
- REM_model.Rmd

Data Directory

1. trans.csv (transcript data)
2. preds.csv (predicted label)
3. perfs.csv
4. nek21.xlsx (Synthesized from the initial 3)
5. speaker table
6. interaction table (for high and low and combined)
7. REM datasets for high & low
8. `surv_object` for high & low
9. `Model_result.Rmd`:model output for high & low and visualization

Setup

The following R packages are necessary to run the REM analysis:

```
if (!require("igraph")) install.packages("igraph")
if (!require("rem")) install.packages("rem")
if (!require("network")) install.packages("network")
if (!require("tidyr")) install.packages("tidyr")
if (!require("caret")) install.packages("caret")
if (!require("survival")) install.packages("survival")
if (!require("dplyr")) install.packages("dplyr")
if (!require("ggplot2")) install.packages("ggplot2")
if (!require("ggraph")) install.packages("ggraph")
if (!require("ggraph")) install.packages("library(ggsurvfit)")
library(knitr)
# do not echo or run code
knitr::opts_chunk$set(echo = FALSE, eval = FALSE)
```

I. Data Preparation & EDA

A. Data Cleaning (`data_clean.Rmd`)

RCode: `data_clean.Rmd`

Input file: nek21.xlsx

Output:

1. filtered_dialog_data.csv (filtered dialogue data based on only main speakers)
2. senders.csv (actor attribute)

Main changes

- Original Data: 8 Senders (“Alex” “Ashley” “Igor” “Katya” “Oleg” “Saleh” “Vika” “Will”)
- After Filtering: 5 Senders (“Ashley”, “Will”, “Saleh”, “Oleg”, “Vika”)

Rationale

- Igor was present only in session 2102
- Katya was present only in the first three sessions: 2102, 2103, 2104
- Alex seemed to be out of nowhere
- Only five ‘consistent’ senders

B. Data Preparation for REM Modeling (`data_process.Rmd`)

Output

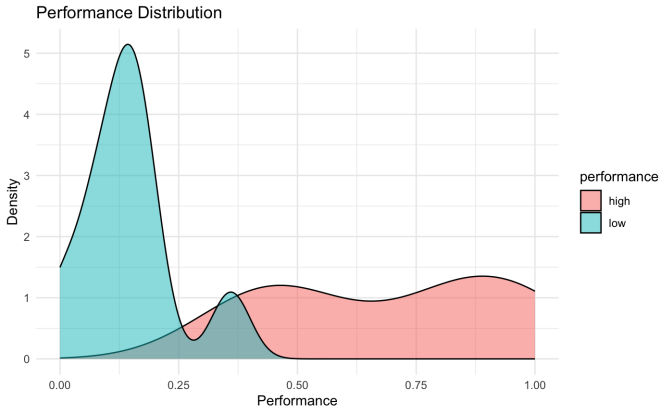
1. high_perf_interactions.RDS
2. low_perf_interactions.RDS
3. high_perf_interactions.csv
4. low_perf_interactions.csv

Data processing deals with three tables

1. **Interactions Data**:
    - Contains information about the sender, receiver, time of interaction, and type of interaction (dialogue act type)
2. **Actors Data**: Contains attributes of each actor, such as name and gender
3. **Performance Data**
    - Min 0.0000
    - Mean 0.4202 0.7149
    - Max 1.0000

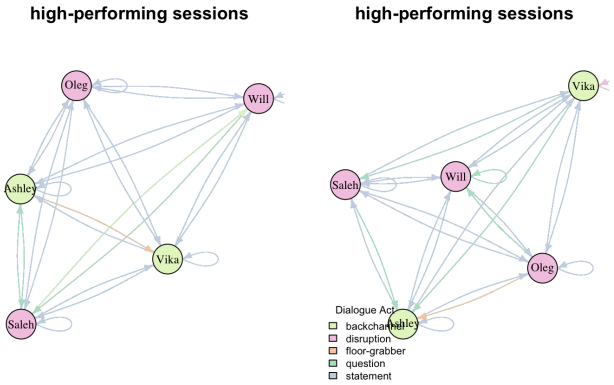
performance	mean
high	0.6965765
low	0.1437967

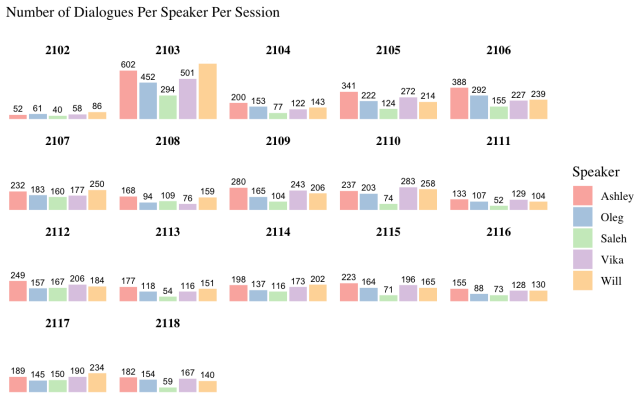
- High: 2117 2111 2113 2107 2110 2116 2106 2102 2118
- Low: 2112 2104 2101 2105 2115 2108 2103 2114 2109



C. Data Visualization (`data_viz.Rmd`)

see `rem_survival_analysis.Rmd` for visualizations





II. REM Modeling (`REM_model.RmD`)

A. Creating REM Dataset

Files/Data Output

1. `REM.data.high.RData`
2. `REM.data.low.RData`

Overview

-

eventID	sender	target	eventTime	eventDummy	eventAtRiskFrom	eventAtRiskUntil	eventAttribute
eventID1	1	2	1	1	1	1	statement
eventID2	1	3	1	0	1	2	statement
eventID2	1	3	2	1	1	2	statement
eventID3	3	2	1	0	1	3	statement
eventID3	3	2	2	0	1	3	statement
eventID3	3	2	3	1	1	3	statement
eventID4	4	1	1	0	1	4	question
eventID4	4	1	2	0	1	4	question
eventID4	4	1	3	0	1	4	question
eventID4	4	1	4	1	1	4	question
eventID5	1	2	1	0	1	5	question
eventID5	1	2	2	0	1	5	question
eventID5	1	2	3	0	1	5	question
eventID5	1	2	4	0	1	5	question

Before fitting a relational event model, data must be prepared in a specific format:

- **EVENTID: Timestamp:** Each interaction event should be time-stamped (in this case 1, 2, 3. . .)
- **Speaker Sequence:** The sequence of speakers or actors involved in each event (same as the timestamp in this case)
- **Performance:** An indicator of the performance level (e.g., high or low) associated with the event (which is how we divided the dataset)
- **Dialogue Act Classification:** Each interaction should be classified according to its dialogue act type (e.g., statement, question, backchannel).

B. Methodology

Incremental Model Building

Files/Data Output

1. `surv_object_high.RData`
2. `surv_object_low.RData`

Overview Incremental model building involves fitting several models to understand the effect of different factors on interaction events:

- **Model 0**: Baseline hazard function.
- **Model 1**: Effect of sender attributes.
- **Model 2**: Effect of receiver attributes.
- **Model 3**: Combined sender and receiver effects.
- **Model 4**: Effect of dialogue act types.
- **Model 5**: Combined sender and dialogue act effects.

C. Comparative Analysis & Result

Files/Data Output

1. `data/high_output.RData`
2. `data/low_output.RData`
3. `Model_result.Rmd`

Useful statistics

```
## Call:
## coxph(formula = highsurv ~ sender + eventAttribute, data = high)
##
##   n= 750599, number of events= 6980
##
##                                coef exp(coef) se(coef)       z Pr(>|z|)
## sender2                    -0.13731   0.87170  0.03472  -3.955 7.66e-05 ***
## sender3                    -0.68108   0.50607  0.04242 -16.056  < 2e-16 ***
## sender4                    -0.27196   0.76188  0.03626  -7.501 6.33e-14 ***
## sender5                    -0.13526   0.87349  0.03542  -3.818 0.000134 ***
## eventAttributedisruption   -0.23283   0.79229  0.10039  -2.319 0.020380 *
## eventAttributefloor-grabber  0.16345   1.17756  0.08691   1.881 0.060030 .
## eventAttributequestion       1.46133   4.31168  0.06942  21.050  < 2e-16 ***
## eventAttributestatement      2.40397  11.06700  0.06652  36.138  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                            exp(coef) exp(-coef) lower .95 upper .95
## sender2                       0.8717    1.14719    0.8144    0.9331
## sender3                       0.5061    1.97602    0.4657    0.5499
## sender4                       0.7619    1.31254    0.7096    0.8180
## sender5                       0.8735    1.14483    0.8149    0.9363
## eventAttributedisruption      0.7923    1.26217    0.6508    0.9646
## eventAttributefloor-grabber   1.1776    0.84921    0.9931    1.3963
## eventAttributequestion        4.3117    0.23193    3.7632    4.9401
## eventAttributestatement      11.0670    0.09036    9.7142   12.6082
##
## Concordance= 0.764  (se = 0.003 )
## Likelihood ratio test= 5704  on 8 df,   p=<2e-16
## Wald test            = 4179  on 8 df,   p=<2e-16
## Score (logrank) test = 5741  on 8 df,   p=<2e-16
```

- Overall model fitness (Concordance, AIC, or BIC)

- Concordance
- Likelihood ratio test
- Wald test = 4179
- Score (logrank)

```
Concordance= 0.764  (se = 0.003 )
Likelihood ratio test= 5704  on 8 df,   p=<2e-16
Wald test            = 4179  on 8 df,   p=<2e-16
Score (logrank) test = 5741  on 8 df,   p=<2e-16
```

- Independent Variables

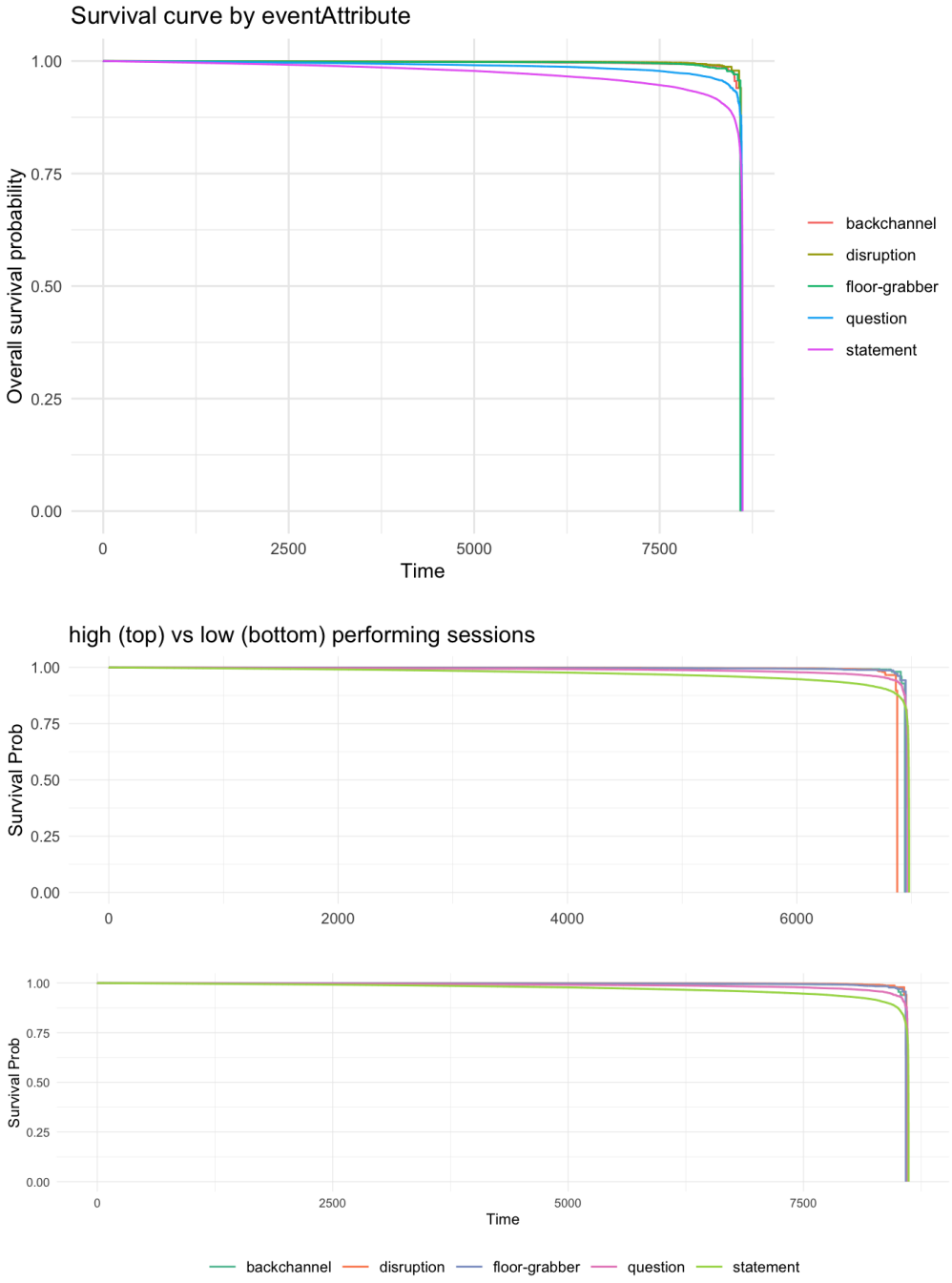
```
## Call:
## coxph(formula = highsurv ~ sender + eventAttribute, data = high)
##
##   n= 750599, number of events= 6980
##
##                                coef exp(coef) se(coef)       z Pr(>|z|)
## sender2                    -0.13731   0.87170  0.03472  -3.955 7.66e-05 ***
## sender3                    -0.68108   0.50607  0.04242 -16.056  < 2e-16 ***
## sender4                    -0.27196   0.76188  0.03626  -7.501 6.33e-14 ***
## sender5                    -0.13526   0.87349  0.03542  -3.818 0.000134 ***
## eventAttributedisruption   -0.23283   0.79229  0.10039  -2.319 0.020380 *
## eventAttributefloor-grabber  0.16345   1.17756  0.08691   1.881 0.060030 .
## eventAttributequestion       1.46133   4.31168  0.06942  21.050  < 2e-16 ***
## eventAttributestatement      2.40397  11.06700  0.06652  36.138  < 2e-16 ***
## ---
```

  - `coef (sign & strength)`
  - `P-value` (statistical sinificance)
  - `exp(coef)`

- Hazard ratio
  - HR > 1: Increases the likelihood of the event. For example, a high hazard ratio for questions indicates that asking questions significantly drives subsequent interactions.
  - HR < 1: Decreases the likelihood of the event.
  - HR $= 1$: No effect on the event likelihood

```
                           exp(coef) exp(-coef) lower .95 upper .95
sender2                       0.8717    1.14719    0.8144    0.9331
sender3                       0.5061    1.97602    0.4657    0.5499
sender4                       0.7619    1.31254    0.7096    0.8180
sender5                       0.8735    1.14483    0.8149    0.9363
eventAttributedisruption      0.7923    1.26217    0.6508    0.9646
eventAttributefloor-grabber   1.1776    0.84921    0.9931    1.3963
eventAttributequestion        4.3117    0.23193    3.7632    4.9401
```

D. Result (Survival) Visualization



TBD

- **Incorporate Degree Centrality:** To understand the influence of key members in triggering interactions.
- **Predict Session Performance:** Using features like participant gender, type and sequence of dialogues, and interaction frequency.
- **Tune Classification Models:** For better accuracy in classifying sessions as high or low performing.