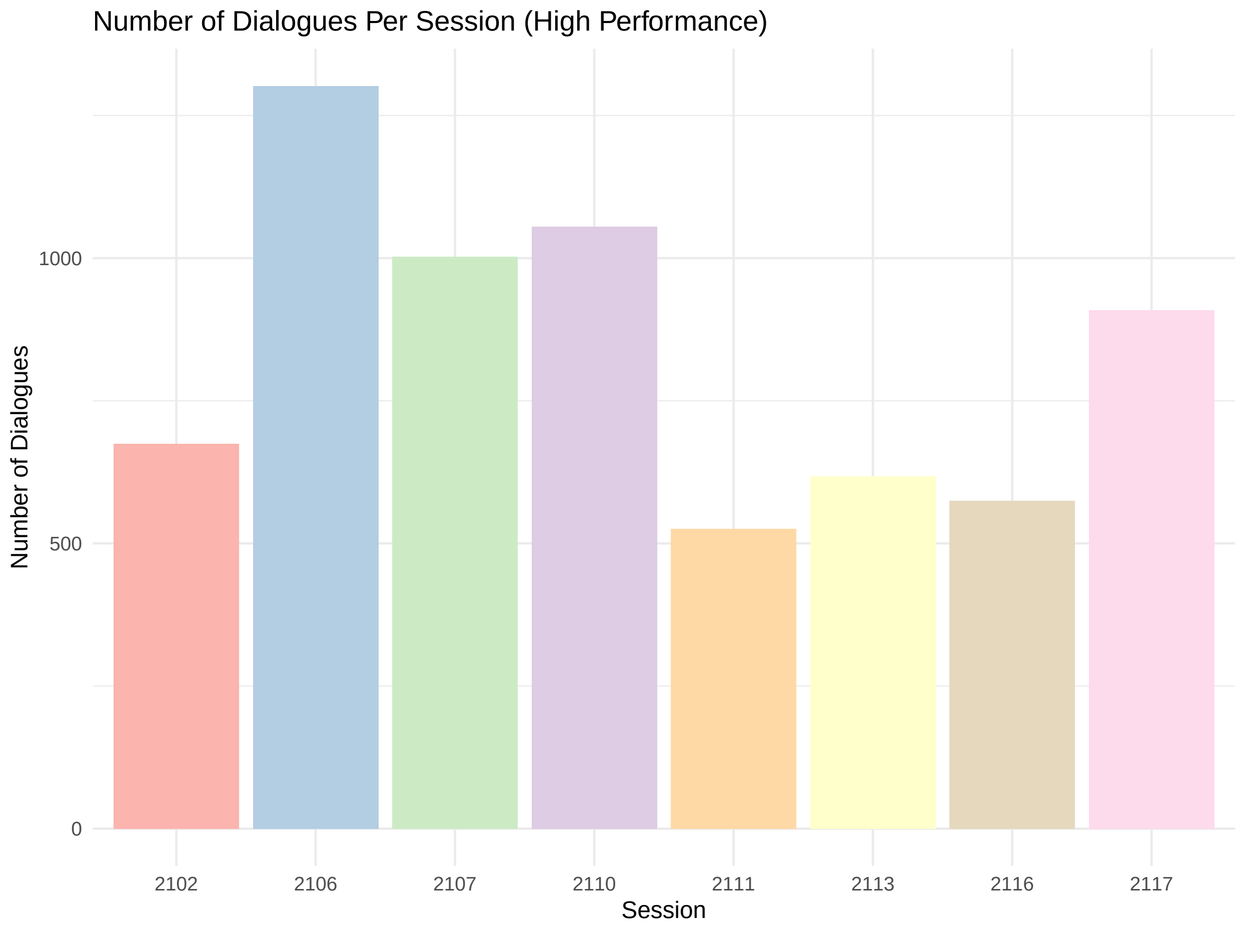
REM for Low Performing Sessions

```r
# Interactions Data Frame (Edges)
low_perf_interactions <- readRDS("data/low_performance_sessions.RData") %>%
  select(session, sender_id, receiver_id, dialog, time)


interactions <- low_perf_interactions %>%
  mutate(
    sender_id = as.integer(sender_id),
    receiver_id = as.integer(receiver_id),
    dialog = as.factor(dialog)
  )



actors_attributes <- data.frame(
  id = 1:8,
  name = c("Igor", "Ashley", "Will", "Katya", "Saleh", "Oleg", "Vika", "Alex"),
  gender = c("male", "female", "male", "female", "male", "male", "female", "male")
)

# Create dummy variables for gender
dummyvars <- dummyVars(" ~ gender", data = actors_attributes)
actors_attributes <- cbind(actors_attributes, predict(dummyvars, actors_attributes)) %>%
  select(id, name, gendermale)
```

Summary by Session and Speaker

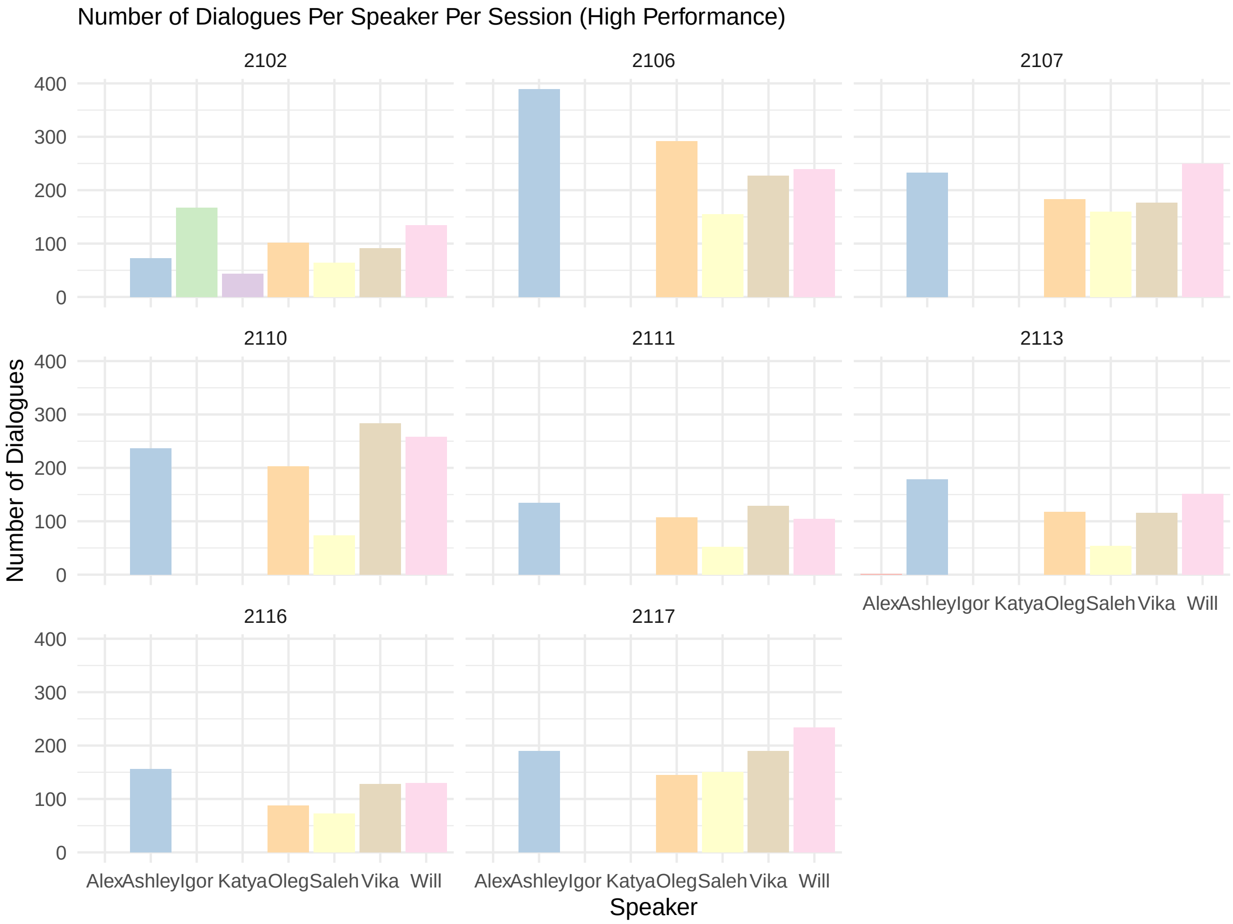
```r
session_dialogues <- low_perf_interactions %>%
  group_by(session) %>%
  summarise(n = n())


ggplot(session_dialogues, aes(x = factor(session), y = n, fill = factor(session))) +
  geom_bar(stat = "identity") +
  scale_fill_brewer(palette = "Pastel1") +
  labs(title = "Number of Dialogues Per Session (High Performance)",
       x = "Session",
       y = "Number of Dialogues",
       fill = "Session") +
  theme_minimal() +
  theme(legend.position = "none")
```



```r
dialogues_per_speaker_session <- low_perf_interactions %>%
  left_join(actors_attributes, by = c("sender_id" = "id")) %>%
  group_by(session, name) %>%
  summarise(number_of_dialogues = n(), .groups = 'drop') %>%
  arrange(session, desc(number_of_dialogues))

dialogues_summary_tibble <- as_tibble(dialogues_per_speaker_session)
print(dialogues_summary_tibble)
```

```
## # A tibble: 43 x 3
##    session name   number_of_dialogues
##      <dbl> <chr>                <int>
##  1    2102 Igor                   167
##  2    2102 Will                   134
##  3    2102 Oleg                   102
##  4    2102 Vika                    91
##  5    2102 Ashley                  73
##  6    2102 Saleh                   64
##  7    2102 Katya                   44
##  8    2106 Ashley                 389
##  9    2106 Oleg                   292
## 10    2106 Will                   239
## # i 33 more rows
```

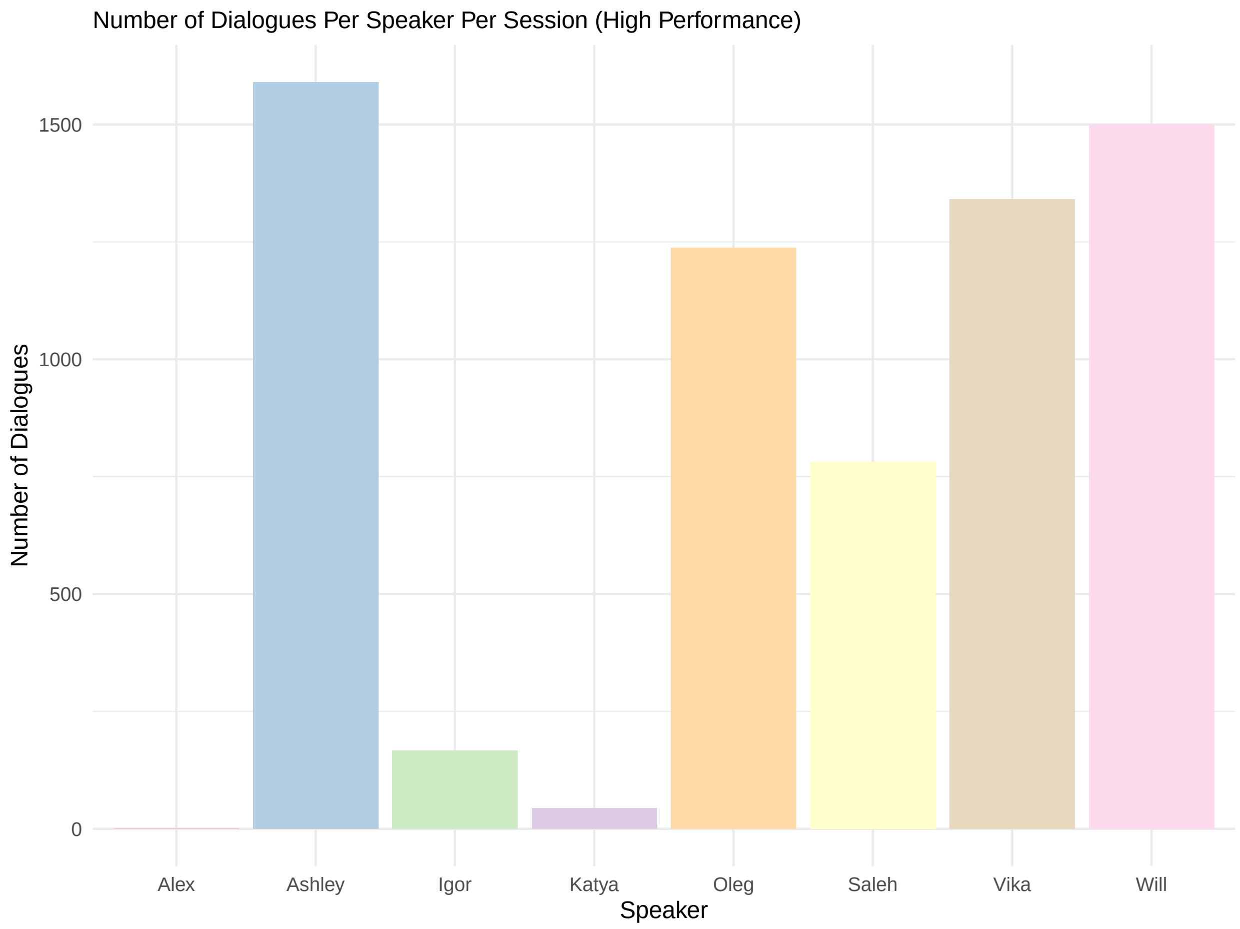
```r
ggplot(dialogues_summary_tibble, aes(x = name, y = number_of_dialogues, fill = name)) +
  geom_bar(stat = "identity") +
  facet_wrap(~session) +
  scale_fill_brewer(palette = "Pastel1") +
  labs(subtitle = "Number of Dialogues Per Speaker Per Session (High Performance)",
       x = "Speaker",
       y = "Number of Dialogues",
       fill = "Speaker") +
  theme_minimal() +
  theme(legend.position = "none")
```



```r
total_dialogues_per_speaker <- low_perf_interactions %>%
  left_join(actors_attributes, by = c("sender_id" = "id")) %>%
  group_by(name) %>%
  summarise(number_of_dialogues = n(), .groups = 'drop') %>%
  arrange(desc(number_of_dialogues)) %>%  as_tibble()
total_dialogues_per_speaker %>% print()
```

```
## # A tibble: 8 x 2
##   name   number_of_dialogues
##   <chr>                <int>
## 1 Ashley                1590
## 2 Will                  1500
## 3 Vika                  1341
## 4 Oleg                  1238
## 5 Saleh                  782
## 6 Igor                   167
## 7 Katya                   44
## 8 Alex                     1
```

```
ggplot(total_dialogues_per_speaker, aes(x = name, y = number_of_dialogues, fill = name)) +
  geom_bar(stat = "identity") +
  scale_fill_brewer(palette = "Pastel1") +
  labs(subtitle = "Number of Dialogues Per Speaker Per Session (High Performance)",
       x = "Speaker",
       y = "Number of Dialogues",
       fill = "Speaker") +
  theme_minimal() +
  theme(legend.position = "none")
```

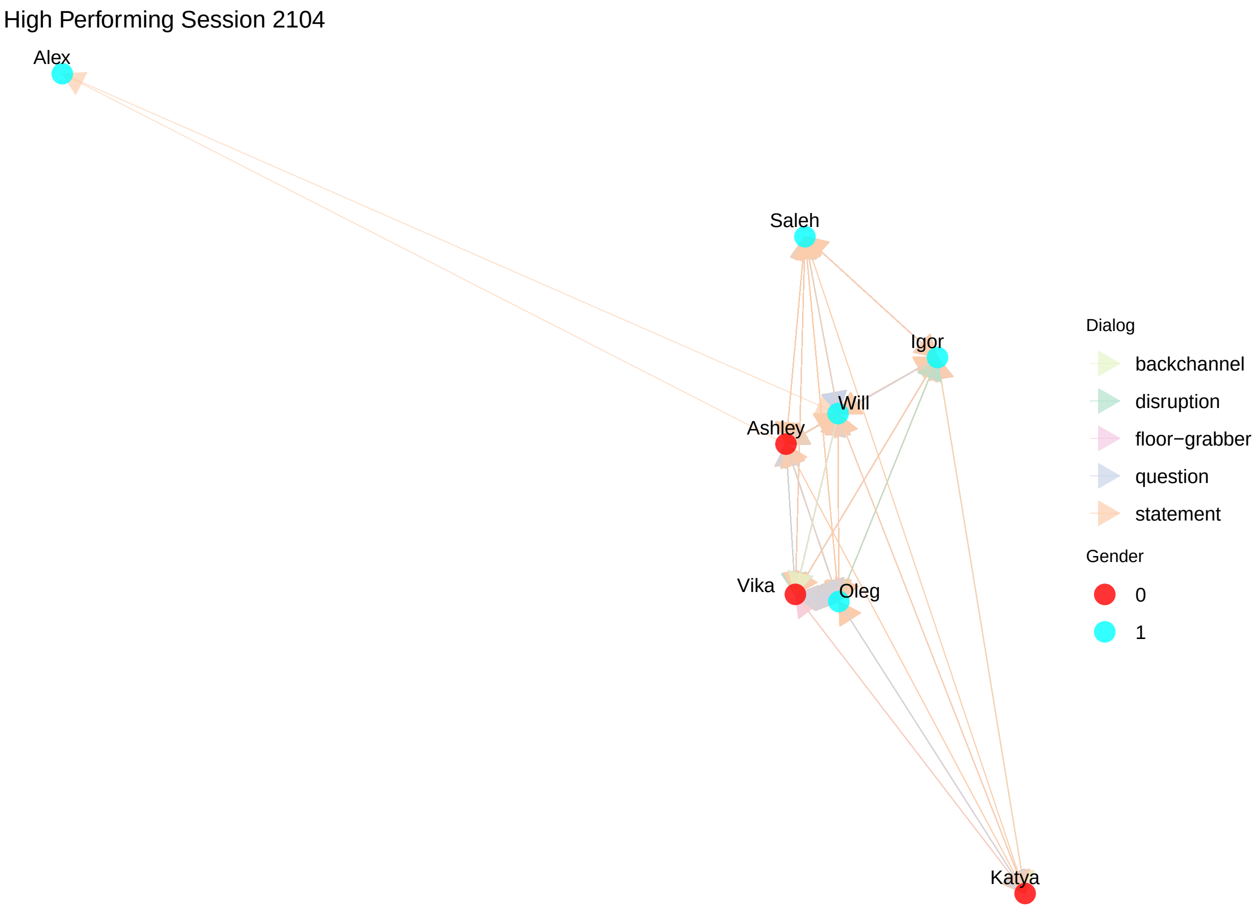


```
dialog_colors <- RColorBrewer::brewer.pal(n = length(unique(interactions$dialog)), name = "Pastel2")
dialog_color_map <- setNames(dialog_colors, unique(interactions$dialog))

low_perf_interactions %>% filter(receiver_id != 0)  %>% select(-session) %>% mutate(time = 1:nrow(.))
```

```
actors_attributes %>% filter(id %in% interactions$sender_id) %>% filter(id %in% interactions$receiver_i
head(actors_attributes)
```

```
##   id   name gendermale
## 1  1   Igor          1
## 2  2 Ashley          0
## 3  3   Will          1
## 4  4  Katya          0
## 5  5  Saleh          1
## 6  6   Oleg          1
```

```
g_subset <- graph_from_data_frame(interactions, directed = TRUE, vertices = data.frame(actors_attribute


V(g_subset)$gender <- actors_attributes$gender[match(V(g_subset)$name, actors_attributes$name)]
V(g_subset)$name <- actors_attributes$name[match(V(g_subset)$name, actors_attributes$name)]
ggraph(g_subset, layout = 'fr') +
  geom_edge_link(aes(color = dialog), alpha = 0.7, edge_width = .2, lineend = "butt", arrow = arrow(typ
  scale_edge_color_manual(values = dialog_color_map) +
  geom_node_point(aes(color = factor(gender)), size = 4, alpha = 0.8) +
  geom_node_text(aes(label = name), repel = TRUE,  color = "black", size = 3, vjust = 1, nudge_x = -.02
  scale_color_manual(values = c('0' = 'red', '1' = 'cyan')) +
  theme_void() +
  labs(subtitle = "High Performing Session 2104", color = "Gender", edge_color = "Dialog") +
  theme(legend.position = "right", legend.title = element_text(size = 8))
```



```
# remove alex from sender_id
actors_attributes %>% filter(name != "Alex") -> actors_attributes

low_perf_interactions %>% filter(receiver_id != 0) %>% select(-session) %>% mutate(time = 1:nrow(.))   %

actors_attributes %>% filter(id %in% interactions$sender_id) %>% filter(id %in% interactions$receiver_i
head(actors_attributes)
```

```
##   id   name gendermale
## 1  1   Igor          1
## 2  2 Ashley          0
## 3  3   Will          1
## 4  4  Katya          0
## 5  5  Saleh          1
## 6  6   Oleg          1
```

```
low_perf_interactions %>% filter(receiver_id != 0) %>% select(-session) %>% mutate(time = 1:nrow(.))   %

dim(interactions)
```

```
## [1] 6655    4
```

```r
head(interactions)
```

```
## # A tibble: 6 x 4
##   sender_id receiver_id  time dialog
##       <int>       <int> <int> <fct>
## 1         1           2     1 disruption
## 2         2           3     2 statement
## 3         3           1     3 question
## 4         1           2     4 statement
## 5         2           1     5 statement
## 6         1           3     6 statement
```

```r
actors_attributes %>% filter(id %in% interactions$sender_id) %>% filter(id %in% interactions$receiver_id

g_subset <- graph_from_data_frame(interactions, directed = TRUE, vertices = data.frame(actors_attributes


V(g_subset)$gender <- actors_attributes$gender[match(V(g_subset)$name, actors_attributes$name)]
V(g_subset)$name <- actors_attributes$name[match(V(g_subset)$name, actors_attributes$name)]
```

```r
interactions$time<-as.numeric(interactions$time)


REM.data <- createRemDataset(
  data = interactions,
  sender = interactions$sender_id,
  target = interactions$receiver_id,
  eventSequence = interactions$time,
  eventAttribute = interactions$dialog,
  atEventTimesOnly = TRUE,
  untilEventOccurrs = TRUE,
  includeAllPossibleEvents = FALSE,
  returnInputData = FALSE
)

#save as RDS
#saveRDS(REM.data, "data/RemDatasetLow.RDS")
```

```r
readRDS("data/REM_data.RDS") -> REM.data
```

```r
dim(REM.data)
```

```
## [1] 90290    12
```

```r
str(REM.data)
```

```
## 'data.frame':    90290 obs. of  12 variables:
##  $ target          : chr  "2" "2" "2" "2" ...
##  $ sender          : chr  "2" "3" "3" "6" ...
##  $ eventID         : chr  "eventID1" "eventID96" "eventID96" "eventID969" ...
##  $ eventTime       : num  1 38 39 959 960 961 962 179 180 181 ...
##  $ eventDummy      : num  1 0 0 0 0 0 0 0 0 0 ...
##  $ eventAtRiskFrom : num  1 1 1 949 949 949 949 1 1 1 ...
##  $ eventAtRiskUntil: num  1 96 96 969 969 969 969 199 199 199 ...
##  $ eventAttribute  : chr  "disruption" "statement" "statement" "statement" ...
##  $ name.x          : chr  "Ashley" "Will" "Will" "Oleg" ...
##  $ gendermale.x    : num  0 1 1 1 1 1 1 0 0 0 ...
##  $ name.y          : chr  "Ashley" "Ashley" "Ashley" "Ashley" ...
##  $ gendermale.y    : num  0 0 0 0 0 0 0 0 0 0 ...
```

```
surv_object <- Surv(time = REM.data$eventTime, event = REM.data$eventDummy)
```

```
base_model <- coxph(surv_object ~ 1, data = REM.data)
summary(base_model)
```

```
## Call:  coxph(formula = surv_object ~ 1, data = REM.data)
##
## Null model
##   log likelihood= -9798.436
##   n= 90290
```

```
sender_model <- coxph(surv_object ~ sender + 1, data = REM.data)


summary(sender_model)
```

```
## Call:
## coxph(formula = surv_object ~ sender + 1, data = REM.data)
##
##   n= 90290, number of events= 986
##
##             coef exp(coef) se(coef)      z Pr(>|z|)
## sender3 -0.32765   0.72062  0.09916 -3.304 0.000952 ***
## sender4 -0.23043   0.79419  0.10508 -2.193 0.028315 *
## sender5 -0.66512   0.51421  0.12440 -5.347 8.96e-08 ***
## sender6 -0.33555   0.71495  0.09907 -3.387 0.000706 ***
## sender7 -0.32023   0.72598  0.10409 -3.076 0.002095 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##         exp(coef) exp(-coef) lower .95 upper .95
## sender3    0.7206      1.388    0.5933    0.8752
## sender4    0.7942      1.259    0.6464    0.9758
## sender5    0.5142      1.945    0.4029    0.6562
## sender6    0.7149      1.399    0.5888    0.8682
## sender7    0.7260      1.377    0.5920    0.8903
##
## Concordance= 0.559  (se = 0.011 )
## Likelihood ratio test= 33.56  on 5 df,   p=3e-06
## Wald test            = 33.07  on 5 df,   p=4e-06
## Score (logrank) test = 33.65  on 5 df,   p=3e-06
```

```
rec_model <- coxph(surv_object ~ target + 1, data = REM.data)


summary(rec_model)
```

```
## Call:
## coxph(formula = surv_object ~ target + 1, data = REM.data)
##
##   n= 90290, number of events= 986
##
##             coef exp(coef) se(coef)      z Pr(>|z|)
## target3 -0.31147   0.73237  0.09916 -3.141  0.00168 **
## target4 -0.05321   0.94818  0.10539 -0.505  0.61366
## target5 -0.58970   0.55449  0.12453 -4.735 2.19e-06 ***
## target6 -0.41861   0.65796  0.09931 -4.215 2.49e-05 ***
## target7 -0.07277   0.92981  0.10402 -0.700  0.48419
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##         exp(coef) exp(-coef) lower .95 upper .95
## target3    0.7324      1.365    0.6030    0.8895
## target4    0.9482      1.055    0.7712    1.1657
## target5    0.5545      1.803    0.4344    0.7078
## target6    0.6580      1.520    0.5416    0.7993
## target7    0.9298      1.075    0.7583    1.1401
##
## Concordance= 0.566  (se = 0.011 )
## Likelihood ratio test= 40.48  on 5 df,   p=1e-07
## Wald test            = 39.3  on 5 df,   p=2e-07
## Score (logrank) test = 39.94  on 5 df,   p=2e-07
```

```
snd_rec_model <- coxph(surv_object ~ sender + target + 1, data = REM.data)
summary(snd_rec_model)
```

```
## Call:
## coxph(formula = surv_object ~ sender + target + 1, data = REM.data)
##
##   n= 90290, number of events= 986
##
##            coef exp(coef) se(coef)      z Pr(>|z|)
## sender3 -0.4268    0.6526   0.1033 -4.131 3.62e-05 ***
## sender4 -0.2163    0.8055   0.1072 -2.018   0.0436 *
## sender5 -0.8191    0.4408   0.1267 -6.463 1.03e-10 ***
## sender6 -0.4809    0.6182   0.1024 -4.697 2.63e-06 ***
## sender7 -0.4295    0.6509   0.1070 -4.015 5.94e-05 ***
## target3 -0.4370    0.6460   0.1033 -4.230 2.34e-05 ***
## target4 -0.1341    0.8745   0.1077 -1.245   0.2131
## target5 -0.7362    0.4789   0.1270 -5.798 6.72e-09 ***
## target6 -0.5767    0.5617   0.1027 -5.618 1.94e-08 ***
## target7 -0.1409    0.8686   0.1053 -1.339   0.1807
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
##         exp(coef) exp(-coef) lower .95 upper .95
## sender3    0.6526      1.532    0.5329    0.7991
## sender4    0.8055      1.241    0.6529    0.9938
## sender5    0.4408      2.268    0.3439    0.5651
## sender6    0.6182      1.618    0.5058    0.7556
## sender7    0.6509      1.536    0.5278    0.8026
## target3    0.6460      1.548    0.5275    0.7910
## target4    0.8745      1.143    0.7081    1.0800
## target5    0.4789      2.088    0.3734    0.6143
## target6    0.5617      1.780    0.4593    0.6869
## target7    0.8686      1.151    0.7067    1.0676
##
## Concordance= 0.606  (se = 0.011 )
## Likelihood ratio test= 93.47  on 10 df,   p=1e-15
## Wald test            = 90.93  on 10 df,   p=4e-15
## Score (logrank) test = 92.43  on 10 df,   p=2e-15
```

```
event_model <- coxph(surv_object ~ eventAttribute + 1, data = REM.data)
summary(event_model)
```

```
## Call:
## coxph(formula = surv_object ~ eventAttribute + 1, data = REM.data)
##
##   n= 90290, number of events= 986
##
##                                  coef exp(coef) se(coef)     z Pr(>|z|)
## eventAttributedisruption    0.1617    1.1755   0.2717 0.595    0.552
## eventAttributefloor-grabber 0.3357    1.3989   0.2273 1.477    0.140
## eventAttributequestion      0.9340    2.5447   0.1853 5.041 4.63e-07 ***
## eventAttributestatement     1.4839    4.4099   0.1789 8.294  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                               exp(coef) exp(-coef) lower .95 upper .95
## eventAttributedisruption        1.176     0.8507    0.6901     2.002
## eventAttributefloor-grabber     1.399     0.7148    0.8961     2.184
## eventAttributequestion          2.545     0.3930    1.7698     3.659
## eventAttributestatement         4.410     0.2268    3.1056     6.262
##
## Concordance= 0.641  (se = 0.009 )
## Likelihood ratio test= 207  on 4 df,   p=<2e-16
## Wald test            = 172.3  on 4 df,   p=<2e-16
## Score (logrank) test = 190.6  on 4 df,   p=<2e-16
```

```
model4 <- coxph(surv_object ~ sender + eventAttribute, data = REM.data)
summary(model4)
```

```
## Call:
## coxph(formula = surv_object ~ sender + eventAttribute, data = REM.data)
##
##   n= 90290, number of events= 986
##
```

```
##                                 coef exp(coef) se(coef)      z Pr(>|z|)
## sender3                     -0.32011   0.72607  0.09953 -3.216 0.001299 **
## sender4                     -0.36260   0.69586  0.10563 -3.433 0.000598 ***
## sender5                     -0.76721   0.46431  0.12467 -6.154 7.55e-10 ***
## sender6                     -0.49032   0.61243  0.09967 -4.919 8.68e-07 ***
## sender7                     -0.32726   0.72089  0.10440 -3.135 0.001720 **
## eventAttributedisruption     0.23939   1.27048  0.27340  0.876 0.381237
## eventAttributefloor-grabber  0.38767   1.47355  0.22795  1.701 0.088995 .
## eventAttributequestion       1.04592   2.84602  0.18777  5.570 2.55e-08 ***
## eventAttributestatement      1.58097   4.85967  0.18048  8.760  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                             exp(coef) exp(-coef) lower .95 upper .95
## sender3                        0.7261     1.3773    0.5974    0.8825
## sender4                        0.6959     1.4371    0.5657    0.8559
## sender5                        0.4643     2.1538    0.3637    0.5928
## sender6                        0.6124     1.6328    0.5038    0.7446
## sender7                        0.7209     1.3872    0.5875    0.8846
## eventAttributedisruption       1.2705     0.7871    0.7434    2.1711
## eventAttributefloor-grabber    1.4735     0.6786    0.9426    2.3035
## eventAttributequestion         2.8460     0.3514    1.9697    4.1122
## eventAttributestatement        4.8597     0.2058    3.4118    6.9220
##
## Concordance= 0.665  (se = 0.01 )
## Likelihood ratio test= 254.2  on 9 df,   p=<2e-16
## Wald test            = 219.7  on 9 df,   p=<2e-16
## Score (logrank) test = 238.7  on 9 df,   p=<2e-16
```

```
model5 <- coxph(surv_object ~ sender * eventAttribute, data = REM.data)
summary(model5)
```

```
## Call:
## coxph(formula = surv_object ~ sender * eventAttribute, data = REM.data)
##
##   n= 90290, number of events= 986
##
##                                         coef exp(coef) se(coef)      z Pr(>|z|)
## sender3                             -0.33080   0.71835  0.50398 -0.656 0.511585
## sender4                              0.87067   2.38851  0.62777  1.387 0.165462
## sender5                              1.46375   4.32215  0.58731  2.492 0.012692
## sender6                              0.79299   2.20999  1.06982  0.741 0.458551
## sender7                              0.23690   1.26731  0.53462  0.443 0.657682
## eventAttributedisruption             0.95970   2.61091  0.47619  2.015 0.043865
## eventAttributefloor-grabber          1.00197   2.72363  0.44237  2.265 0.023513
## eventAttributequestion               1.43755   4.21037  0.40317  3.566 0.000363
## eventAttributestatement              1.95909   7.09285  0.38693  5.063 4.13e-07
## sender3:eventAttributedisruption    -0.06494   0.93712  0.81898 -0.079 0.936798
## sender4:eventAttributedisruption    -1.88494   0.15184  0.98907 -1.906 0.056680
## sender5:eventAttributedisruption    -3.83539   0.02159  1.19637 -3.206 0.001347
## sender6:eventAttributedisruption    -1.29988   0.27257  1.24935 -1.040 0.298136
## sender7:eventAttributedisruption    -1.33482   0.26321  0.93241 -1.432 0.152266
## sender3:eventAttributefloor-grabber -0.76328   0.46614  0.68799 -1.109 0.267247
## sender4:eventAttributefloor-grabber -1.35756   0.25729  0.83456 -1.627 0.103807
## sender5:eventAttributefloor-grabber -2.16281   0.11500  0.77238 -2.800 0.005107
## sender6:eventAttributefloor-grabber -1.22900   0.29259  1.15728 -1.062 0.288248
## sender7:eventAttributefloor-grabber -1.07158   0.34247  0.71085 -1.507 0.131691
## sender3:eventAttributequestion       0.30547   1.35727  0.54708  0.558 0.576595
## sender4:eventAttributequestion      -1.15034   0.31653  0.65948 -1.744 0.081104
## sender5:eventAttributequestion      -1.92038   0.14655  0.63087 -3.044 0.002334
## sender6:eventAttributequestion      -1.31403   0.26874  1.08797 -1.208 0.227133
## sender7:eventAttributequestion      -1.02853   0.35753  0.58170 -1.768 0.077039
## sender3:eventAttributestatement      0.02038   1.02059  0.51866  0.039 0.968650
## sender4:eventAttributestatement     -1.27119   0.28050  0.64208 -1.980 0.047726
## sender5:eventAttributestatement     -2.40823   0.08997  0.61037 -3.946 7.96e-05
## sender6:eventAttributestatement     -1.27692   0.27889  1.07713 -1.185 0.235824
## sender7:eventAttributestatement     -0.32597   0.72183  0.54960 -0.593 0.553109
##
## sender3
## sender4
## sender5                             *
## sender6
## sender7
## eventAttributedisruption            *
## eventAttributefloor-grabber         *
## eventAttributequestion              ***
## eventAttributestatement             ***
## sender3:eventAttributedisruption
## sender4:eventAttributedisruption    .
## sender5:eventAttributedisruption    **
## sender6:eventAttributedisruption
## sender7:eventAttributedisruption
## sender3:eventAttributefloor-grabber
## sender4:eventAttributefloor-grabber
## sender5:eventAttributefloor-grabber **
## sender6:eventAttributefloor-grabber
## sender7:eventAttributefloor-grabber
## sender3:eventAttributequestion
## sender4:eventAttributequestion      .
## sender5:eventAttributequestion      **
## sender6:eventAttributequestion
## sender7:eventAttributequestion      .
## sender3:eventAttributestatement
## sender4:eventAttributestatement     *
## sender5:eventAttributestatement     ***
## sender6:eventAttributestatement
## sender7:eventAttributestatement
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##                                     exp(coef) exp(-coef) lower .95 upper .95
## sender3                               0.71835     1.3921   0.26751    1.9290
## sender4                               2.38851     0.4187   0.69787    8.1749
## sender5                               4.32215     0.2314   1.36703   13.6654
## sender6                               2.20999     0.4525   0.27149   17.9896
## sender7                               1.26731     0.7891   0.44444    3.6137
## eventAttributedisruption              2.61091     0.3830   1.02674    6.6393
## eventAttributefloor-grabber           2.72363     0.3672   1.14446    6.4818
## eventAttributequestion                4.21037     0.2375   1.91047    9.2790
## eventAttributestatement               7.09285     0.1410   3.32247   15.1419
## sender3:eventAttributedisruption      0.93712     1.0671   0.18822    4.6657
## sender4:eventAttributedisruption      0.15184     6.5860   0.02185    1.0551
## sender5:eventAttributedisruption      0.02159    46.3116   0.00207    0.2253
## sender6:eventAttributedisruption      0.27257     3.6688   0.02355    3.1544
## sender7:eventAttributedisruption      0.26321     3.7993   0.04233    1.6367
## sender3:eventAttributefloor-grabber   0.46614     2.1453   0.12103    1.7953
## sender4:eventAttributefloor-grabber   0.25729     3.8867   0.05012    1.3207
## sender5:eventAttributefloor-grabber   0.11500     8.6955   0.02531    0.5226
## sender6:eventAttributefloor-grabber   0.29259     3.4178   0.03028    2.8270
## sender7:eventAttributefloor-grabber   0.34247     2.9200   0.08502    1.3794
## sender3:eventAttributequestion        1.35727     0.7368   0.46450    3.9659
## sender4:eventAttributequestion        0.31653     3.1593   0.08691    1.1528
## sender5:eventAttributequestion        0.14655     6.8236   0.04256    0.5046
## sender6:eventAttributequestion        0.26874     3.7211   0.03186    2.2668
## sender7:eventAttributequestion        0.35753     2.7969   0.11433    1.1181
## sender3:eventAttributestatement       1.02059     0.9798   0.36929    2.8206
## sender4:eventAttributestatement       0.28050     3.5651   0.07969    0.9873
## sender5:eventAttributestatement       0.08997    11.1143   0.02720    0.2976
## sender6:eventAttributestatement       0.27889     3.5856   0.03377    2.3030
## sender7:eventAttributestatement       0.72183     1.3854   0.24582    2.1196
##
## Concordance= 0.675  (se = 0.01 )
## Likelihood ratio test= 300  on 29 df,   p=<2e-16
## Wald test            = 238.4  on 29 df,   p=<2e-16
## Score (logrank) test = 282.6  on 29 df,   p=<2e-16
```

```r
model6 <- coxph(surv_object ~ sender + target + eventAttribute, data = REM.data)
summary(model6)
```

```
## Call:
## coxph(formula = surv_object ~ sender + target + eventAttribute,
##     data = REM.data)
##
##   n= 90290, number of events= 986
##
##                                coef exp(coef) se(coef)      z Pr(>|z|)
## sender3                     -0.3765    0.6863   0.1019 -3.693 0.000221 ***
## sender4                     -0.3611    0.6969   0.1074 -3.361 0.000776 ***
## sender5                     -0.9195    0.3987   0.1267 -7.255 4.01e-13 ***
## sender6                     -0.5912    0.5537   0.1017 -5.813 6.15e-09 ***
## sender7                     -0.4166    0.6593   0.1062 -3.922 8.79e-05 ***
## target3                     -0.3773    0.6857   0.1024 -3.684 0.000229 ***
## target4                     -0.2556    0.7745   0.1074 -2.379 0.017382 *
## target5                     -0.8219    0.4396   0.1264 -6.502 7.94e-11 ***
## target6                     -0.5644    0.5687   0.1021 -5.528 3.24e-08 ***
## target7                     -0.2553    0.7747   0.1059 -2.410 0.015947 *
## eventAttributedisruption     0.2084    1.2317   0.2743  0.760 0.447459
## eventAttributefloor-grabber  0.4184    1.5195   0.2292  1.825 0.068002 .
## eventAttributequestion       1.0573    2.8785   0.1886  5.607 2.06e-08 ***
## eventAttributestatement      1.5917    4.9121   0.1815  8.769  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
##                             exp(coef) exp(-coef) lower .95 upper .95
## sender3                        0.6863     1.4572    0.5620    0.8380
## sender4                        0.6969     1.4350    0.5646    0.8602
## sender5                        0.3987     2.5080    0.3110    0.5112
## sender6                        0.5537     1.8062    0.4536    0.6758
## sender7                        0.6593     1.5167    0.5354    0.8119
## target3                        0.6857     1.4584    0.5610    0.8381
## target4                        0.7745     1.2912    0.6274    0.9560
## target5                        0.4396     2.2748    0.3431    0.5632
## target6                        0.5687     1.7584    0.4656    0.6947
## target7                        0.7747     1.2908    0.6295    0.9534
## eventAttributedisruption       1.2317     0.8119    0.7195    2.1085
## eventAttributefloor-grabber    1.5195     0.6581    0.9695    2.3814
## eventAttributequestion         2.8785     0.3474    1.9891    4.1655
## eventAttributestatement        4.9121     0.2036    3.4416    7.0109
##
## Concordance= 0.689  (se = 0.01 )
## Likelihood ratio test= 312.2  on 14 df,   p=<2e-16
## Wald test            = 278.4  on 14 df,   p=<2e-16
## Score (logrank) test = 298.6  on 14 df,   p=<2e-16
```