Frequency Analyses

Introduction

This report identifies markers of politeness in video game dialogue and tests gender differences in the extent and type of politeness strategies.

During conversation, speakers use various linguistic strategies to avoid "face-threatening" acts: utterances that threaten an individual's independence (like making a demand) or their desire to be liked (like insulting them, Brown & Levinson, 1987). One strategy is 'hedging', the use of linguistic markers that affect the epistemic certainty of the speaker's claims. For example, "Maybe we should go to the shops" is less face-threatening than "We should go to the shops", because it hedges the direct demand on a person's time and provides the interlocutor the ability to suggest a different course of action without directly rejecting the speaker.

Greater use of politeness markers such as hedging has been associated with female speech (Lakoff, 1973; Fishman, 1983; Coates 2003; Holmes, 2013; Mirzapour, 2016). The classic divide is between theories that see female hedging as a form of submissiveness (e.g. Lakoff, 1973), and theories that see it as expressing affiliation (Holmes, 1990; Dixon & Foster, 1997). Various studies also suggest that power relations can trump gender relations (e.g. Mullany, 2004). However, in general, both theories predict that females use more hedging. Empirical studies have supported this in the dialogue of female characters in fiction (Karlsson Nordqvist, 2013; Jan & Rahman, 2020; Weisi, & Asakereh, 2021) though there may be differences in the distribution of particular hedges (Holmes, 1990), and there are also studies that find no significant difference between genders (Nemati & Bayer, 2007; Vold, 2006; Holtgraves & Lasky, 1999). Similar strategies for avoiding face-threatening acts include showing gratitude, polite requests (e.g. use of "please"), and apologising (see Danescu-Niculescu-Mizil, 2013). For video games, we would predict greater use of politeness strategies by female characters compared to male characters.

Polite speakers may also aim to avoid the use of negative words and swearing (Jay & Janschewitz, 2008). There are folk beliefs in Western society that men swear more than women (Coates, 2004), and judgements of the acceptability of swearing vary by the gender of the speaker, the interlocutors, and the context (Mills, 2004; DeFrank, & Kahlbaugh, 2019). However, empirical studies of real conversation show mixed relationships between gender and swearing. Some show that men swear more than women (McEnery & Xiao, 2004), some find no overall difference (Baker, 2014; McEnery, 2006), and others find differences are more based on context, age, and specific swear words (Allan & Burridge, 2006; Gauthier & Guille, 2017).

This makes predictions for the video game dialogue difficult. There are surprisingly few studies of swearing by gender in fiction. Cressman (2009) find that men swear more than women in film, and Coyne (2012) found that male characters used more profanity than female characters in adolescent literature, but only for adult characters. There are also no openly-available corpora that have dialogue from fiction which is tagged for gender at the utterance level. However, based on findings in other parts of the study (female characters have more limited roles, are more likely to have neutral emotions, and less likely to be angry), we would predict that female dialogue in video games includes fewer swear words than male dialogue.

Methods

Politeness strategies are identified automatically using 'Convokit' (Chang et al., 2020, see http://convokit.cornell.edu/). This uses machine learning methods trained on a tagged dataset to count the number of cases of various types of politeness strategy. See the python script `analysis/Analyse_Politeness.py`.

An alternative method was used to detect hedging, from Knight, Adolphs & Carter (2013). They obtain frequencies of a list of key phrases. Here we replicate their method using the R package 'Quanteda' (Benoit et al., 2018).

We compare frequencies using the log likelihood measure (G2, see Dunning et al., 1993; Rayson et al., 2004), as used by e.g. the Lancaster Log-likelihood and effect size calculator, https://ucrel.lancs.ac.uk/llwizard.html.

Load libraries

```
library(quanteda)
library(quanteda.textstats)
library(stringr)
library(rjson)
```

Functions to run log likelihood tests according to the G2 measure.

```
logLikelihood.G2 = function(a,b,c,d){
  c = as.double(c)
  d = as.double(d)
  E1 = c*(a+b) / (c+d)
  E2 = d*(a+b) / (c+d)
  G2 = 2*((a*log(a/E1)) + (b*log(b/E2)))
  return(G2)
}

logLikelihood.test = function(freqInCorpus1, freqInCorpus2, sizeOfCorpus1, sizeOfCorpus2){
  # A single test is done like this:
  # logLikelihood.test(2554, 3468, 110000, 140000)
  G2 = logLikelihood.G2(freqInCorpus1,freqInCorpus2,sizeOfCorpus1,sizeOfCorpus2)
  p.value = pchisq(G2, df=2, lower.tail=FALSE)
  #print(paste("Log Likelihood =",G2, ", p = ",p.value))
  return(data.frame(G2 = G2, p = p.value))
}
```

Load data

Load all texts, split into male and female dialouge, tokenise, and count the total number of words.

```
# Number of lines from mini-sample
miniSampleSize = 1000

textF = c()
textM = c()
textFMini = c()
textMMini = c()
stats = read.csv("../results/generalStats.csv",stringsAsFactors = F)
# Remove alternative measures
stats = stats[stats$alternativeMeasure!="True",]
stats = stats[!is.na(stats$words),]
folders = unique(stats$folder)
for(folder in folders){
  dx = fromJSON(file = paste0(folder,"data.json"))["text"]
  dx = unlist(dx)
  names(dx) = gsub("CHOICE\\.","",names(dx))
  names(dx) = gsub("text\\.","",names(dx))
  js = fromJSON(file = paste0(folder,"meta.json"))

  flines = dx[names(dx) %in% js$characterGroups[["female"]]]
  mlines = dx[names(dx) %in% js$characterGroups[["male"]]]

  textF = c(textF, flines)
  textM = c(textM, mlines)
```

```
  firstLines = dx[names(dx) %in% c(
    js$characterGroups[["male"]],
    js$characterGroups[["female"]])]
  if(length(firstLines)>=miniSampleSize){
    firstLines = firstLines[1:miniSampleSize]
      flinesMini = firstLines[names(firstLines) %in% js$characterGroups[["female"]]]
      mlinesMini = firstLines[names(firstLines) %in% js$characterGroups[["male"]]]

      textFMini = c(textFMini, flinesMini)
      textMMini = c(textMMini, mlinesMini)
  }
}
```

Create data frames, convert to the Quanteda corpus format, tokenise and count the number of words in each sub-corpus:

```
dM = data.frame(
  text = textM,
  group="male",stringsAsFactors = F
)
corpM = corpus(dM)
tokensM = tokens(corpM, remove_punct = TRUE)
maleTotal = sum(ntoken(tokensM))

dF = data.frame(
  text = textF,
  group="female",stringsAsFactors = F
)
corpF = corpus(dF)
tokensF = tokens(corpF, remove_punct = TRUE)
femaleTotal = sum(ntoken(tokensF))
```

Combine male and female corpora into one corpus, tagged with their group:

```
d = rbind(dM,dF)
corpAll = corpus(d)
tokensAll = tokens(corpAll, remove_punct = TRUE)
```

Keyness

The target group is the male corpus and the reference group is the female corpus.

```
dfmat <- dfm(tokensAll)
k = textstat_keyness(dfmat,
        target = dfmat$group=="male",
        measure = "lr",sort = F)
k$maleFreqPerMillion = (k$n_target/maleTotal) * 1000000
k$femaleFreqPerMillion = (k$n_reference/femaleTotal) * 1000000

top = k[order(k$G2,decreasing = T),][1:30,]
bottom = k[order(k$G2,decreasing = F),][1:30,]
```

Top words used more by men than women:

```
knitr::kable(top[,c("feature",
        "maleFreqPerMillion","femaleFreqPerMillion","G2")],
         digits = 0,row.names = F)
```

feature	maleFreqPerMillion	femaleFreqPerMillion	G2
alexander	403	38	839
kupo	377	42	735
ya	314	94	311
yeah	897	530	242
ain't	262	84	239
ye	220	68	209
the	43339	40796	208
em	300	122	194
got	1879	1388	189
eh	230	87	166
ah	589	349	157
gotta	313	147	153
hey	894	601	148
uh	326	161	144
dude	56	3	141
shit	126	36	130
cassima	50	3	123
yer	99	24	122
thou	113	32	120
gonna	566	358	119
yo	44	2	117
hell	280	143	115
sora	176	72	114
alexander's	43	2	108
aye	229	113	102
king	444	277	100
riku	68	13	99
noct	53	7	95
no	4587	4029	93
goin	82	22	89

Top words used more by women than by men:

```
knitr::kable(bottom[,c("feature",
         "maleFreqPerMillion","femaleFreqPerMillion","G2")],
          digits = 0,row.names = F)
```

feature	maleFreqPerMillion	femaleFreqPerMillion	G2
i	27144	30322	-472
husband	29	161	-286
he	3866	4696	-212
thank	727	1104	-207
geth	265	504	-203
mother	224	446	-201
flemeth	2	58	-200
crono	26	118	-179
cloud	130	295	-178
she	1915	2457	-178
skipper	1	51	-175
ajira	0	44	-174
oh	1416	1863	-161
father	282	485	-145
please	801	1118	-139
um	115	245	-130
giggle	1	39	-130
narrating	10	65	-126
think	2153	2621	-121
noel	16	76	-119
inmate	1	36	-116
benezia	7	51	-109
niket	0	28	-108
so	4155	4745	-103
cerberus	197	337	-100
griff	1	30	-99
child	111	220	-98
jeff	1	27	-97
yunie	0	23	-96
habasi	0	25	-96

Write out:

```
write.csv(rbind(top,bottom),"../results/keyness.csv")
```

Politeness

Load the politeness measures, calcualted in `analysis/Analyse_Politeness.py`:

```
pol = read.csv("../results/politeness.csv",stringsAsFactors = F)

politeness = NULL
for(feature in unique(pol$feature)){
  nF = pol[pol$group=="female" & pol$feature==feature,]$count
  nM = pol[pol$group=="male" & pol$feature==feature,]$count
  propF = 1000000 * (nF/femaleTotal)
  propM = 1000000 * (nM/maleTotal)
  ll = logLikelihood.test(nM,nF,maleTotal, femaleTotal)
  politeness = rbind(politeness, data.frame(
    feature=feature, nFemale = nF,nMale = nM,
    nFemalePerMillionWords = propF,
    nMalePerMillionWords = propM,
    G2 = ll[1],
    p = ll[2]
  ))
}
```

Adjust p-value for multiple comparisons:

```
politeness$p = p.adjust(politeness$p,method = "bonferroni")
```

Results:

```
knitr::kable(politeness, digits = 2)
```

feature	nFemale	nMale	nFemalePerMillionWords	nMalePerMillionWords	G2	p
Please	763	1042	381.57	282.24	39.30	0.00
Please_start	1320	1738	660.12	470.75	84.06	0.00
HASHEDGE	31160	52489	15582.88	14217.12	163.18	0.00
Indirect_(btw)	77	150	38.51	40.63	0.15	1.00
Hedges	11636	18263	5819.07	4946.70	185.08	0.00
Factuality	4182	6461	2091.39	1750.02	79.50	0.00
Deference	1751	3492	875.66	945.84	6.99	0.64
Gratitude	3218	4487	1609.30	1215.34	145.03	0.00
Apologizing	2157	3077	1078.70	833.43	82.90	0.00
1st_person_pl.	30883	59572	15444.35	16135.62	39.17	0.00
1st_person	44732	78761	22370.13	21333.13	63.99	0.00
1st_person_start	42191	67291	21099.39	18226.38	548.98	0.00
2nd_person	61570	113053	30790.68	30621.43	1.21	1.00
2nd_person_start	14054	27488	7028.30	7445.37	31.09	0.00
Indirect_(greeting)	1707	3773	853.66	1021.95	38.88	0.00
Direct_question	10733	20667	5367.49	5597.84	12.53	0.04
Direct_start	13949	25928	6975.79	7022.83	0.41	1.00
HASPOSITIVE	77780	142014	38897.18	38465.78	6.24	0.93
HASNEGATIVE	61993	116566	31002.22	31572.96	13.49	0.02
SUBJUNCTIVE	600	1009	300.06	273.30	3.26	1.00
INDICATIVE	720	1202	360.07	325.57	4.53	1.00

Summarise results and write to stats:

```
getStatText = function(feature,femaleDiff,G2,pval,w=F){
  diffx = "+"
  if(femaleDiff<0){
    diffx = ""
```

```
  }
  diffx = paste0(diffx,
                 round(100*femaleDiff),
                 "\\%")
  p = pval
  if(p < 0.001){
    p = "< 0.001"
  } else{
    p = paste("=",round(p,3))
  }
  statText = paste0(diffx,", G2 = ",round(G2,2),", p ",p)
  if(w){
    cat(statText, file=paste0(
      "../results/latexStats/Freq_",feature,
      ".tex"))
  }
  return(statText)
}

getStat = function(X,feature,w=F){
  px = X[X$feature==feature,]

  femaleDiff = (px$nFemalePerMillionWords -
                 px$nMalePerMillionWords) /
                px$nMalePerMillionWords
  statText = getStatText(feature,femaleDiff,px$G2,px$p,w)
  return(statText)
}
```

Many of the results are compatible with females exhibiting more frequent politeness strategies than males. For example, compared to male characters, female characters use:

- More hedging (+18%, G2 = 185.08, $p < 0.001$)
- More gratitude (+32%, G2 = 145.03, $p < 0.001$)
- More apologies (+29%, G2 = 82.9, $p < 0.001$)
- More 'please' (+35%, G2 = 39.3, $p < 0.001$)

No significant differences for:

- Direct questions (-4%, G2 = 12.53, $p = 0.04$)
- Negative words (-2%, G2 = 13.49, $p = 0.025$)

Alternative measure of hedging

Knight, Adolphs & Carter (2013) use a different approach to quantifying hedging. They obtain frequencies of a list of key phrases. Here we replicate their method using Quanteda. Two different methods are used to obtain frequency, one for single-word key phrases, and one for multi-word key phrases.

```
hedges = c(
  "Actually", "Generally", "Likely",
  "Only",  "Really", "Surely",
  "Apparently", "Guess", "Maybe",
  "Partially", "Relatively", "Thing",
  "Arguably", "Necessarily", "Possibility",
  "Possibly", "Roughly", "Typically",
  "Broadly", "Just", "Normally",
  "Probably", "Seemingly", "Usually",
  "Frequently", "Quite"
)

hedgePhrases = c(
  "I think",
  "Kind of",
  "Of course",
  "Sort of",
  "You know")

dfmat <- dfm(tokensAll)
dfmat <- dfm_select(dfmat,pattern=hedges)
freq.hedges = textstat_keyness(
  dfmat,target = dfmat$group=="male", measure = "lr",
     sort = F, correction = "none")
ll = logLikelihood.test(freq.hedges$n_target,freq.hedges$n_reference,maleTotal,femaleTotal)
freq.hedges$G2 = ll$G2
freq.hedges$p = ll$p
freq.hedges$maleFreqPerMillion = (freq.hedges$n_target/maleTotal) * 1000000
freq.hedges$femaleFreqPerMillion = (freq.hedges$n_reference/femaleTotal) * 1000000

getPhraseFrequency = function(w,group){
  corp = corpus(d[d$group==group,])
  # We don't want punctuation between phrase parts
  toks = tokens(corp, remove_punct = FALSE)
  k = kwic(toks, pattern = phrase(c(w)))
  length(k$post)
}

for(w in hedgePhrases){
  freqF = getPhraseFrequency(w,"female")
  freqM = getPhraseFrequency(w,"male")
  ll = logLikelihood.test(freqM, freqF, maleTotal, femaleTotal)
  freq.hedges = rbind(freq.hedges, data.frame(
    feature = w,
    G2 = ll[1],
    p = ll[2],
    n_target = freqM,
    n_reference = freqF,
    maleFreqPerMillion = (freqM/maleTotal) * 1000000,
    femaleFreqPerMillion = (freqF/femaleTotal) * 1000000
  ))
}
```

```r
freq.hedges$sig = freq.hedges$p<0.05
freq.hedges[freq.hedges$sig,]
```

```
##      feature        G2            p n_target n_reference maleFreqPerMillion
## 1      guess 12.413760 2.015516e-03     2346        1119         635.435353
## 5      quite  6.074512 4.796634e-02     1755        1047         475.357649
## 7     really 64.708051 8.888434e-15     4142        2739        1121.898224
## 8   actually 11.392939 3.357799e-03     1032         662         279.526549
## NA      <NA>        NA           NA       NA          NA                 NA
## 21 seemingly  6.113577 4.703852e-02       17           2           4.604604
## 25 partially  6.211820 4.478375e-02        8          13           2.166872
## 26   I think 91.206243 1.566087e-20     2584        1875         699.899809
## 27   Kind of 10.606339 4.975799e-03     1098         697         297.403247
##    femaleFreqPerMillion  sig
## 1            559.603247 TRUE
## 5            523.596604 TRUE
## 7           1369.752719 TRUE
## 8            331.061081 TRUE
## NA                   NA   NA
## 21             1.000185 TRUE
## 25             6.501199 TRUE
## 26           937.673001 TRUE
## 27           348.564310 TRUE
```

```r
names(freq.hedges)[names(freq.hedges)=="n_target"] = "freqMale"
names(freq.hedges)[names(freq.hedges)=="n_reference"] = "freqFemale"

hedgeMPerMillion = sum(freq.hedges$freqMale)/maleTotal * 1000000
hedgeFPerMillion = sum(freq.hedges$freqFemale)/femaleTotal * 1000000

ll = logLikelihood.test(sum(freq.hedges$freqMale),
                   sum(freq.hedges$freqFemale),
                   maleTotal,femaleTotal)

femaleDiffHedge = (hedgeFPerMillion - hedgeMPerMillion) /
                hedgeMPerMillion
```

As with the main method, females are more likely to use hedging (male frequency per million = 1.316×10^4, female frequency per million = 1.397×10^4, +6%, G2 = 62.83, p < 0.001)

Swearing

Using keywords identified in TV show dialogue from Bednarek (2019).

```
swears = read.csv("https://gist.githubusercontent.com/tjrobinson/2366772/raw/97329ead3d5ab06160c3c7a
                  stringsAsFactors = F,header = F)
swears = swears[,1]
swears = swears[!swears %in% c("snatch")]
nx = c("hell","dago","ass")
swears[!swears %in% nx]= paste0(swears[!swears %in% nx],"*")

# Specific to games in the corpus
swears = c(swears,"vermin","scum")

# Bedarek
bednarekSwears = c("god",
  "hell",  "damn", "crap", "screw",
  "fuck", "fucktard", "fuckwad", "fucks", "fucking", "butt-fuck",
  "butt-fucking", "fuck-up", "fuckable", "fucked", "pencil-fucked",
  "fucked-up", "ass-fucked", "fucker",
  "motherfucker", "motherfuckers", "motherfucking",
  "bullshit", "dipshit", "shit", "shit-faced", "shit-ass",
  "shitheads", "shits", "shittiest", "shitting", "shitty",
  "damned", "fricking","freaking","frigging",
  "gosh", "heck", "jeez", "shucks")

swears = c(swears,bednarekSwears)
swears = dictionary(list(swears=swears))

dfmat_swears = dfm(tokensAll)
dfmat_swears = dfm_select(dfmat_swears, pattern=swears)
tstat_freq_swears <- textstat_frequency(dfmat_swears, groups = d$group)

swearFreq = tapply(tstat_freq_swears$frequency,tstat_freq_swears$group,sum)
swearFreqPerMillion = (swearFreq / c(femaleTotal,maleTotal)) * 1000000

femaleDiff = (swearFreqPerMillion["female"] -
                 swearFreqPerMillion["male"]) /
                swearFreqPerMillion["male"]

llSwear = logLikelihood.test(swearFreq["male"],
                   swearFreq["female"],
                   maleTotal,femaleTotal)
```

Female characters swear less than male characters (-37%, G2 = 414.06, $p < 0.001$)

Hesitations

```
hesitations = c("um","umm","ummm","ummmm","ummmm","ummmmmm",
                'er','err','errr','errrr',
                "uh","uhh","uhhh", "uhhhh","uhhhhhh",
                "uhhhhhhhhhhhh","uuh","uuuh",
                "uuuuh", "uuuuhh", "uuuuuuuh",
                "ur","ur","urrr")

dfmat_hes = dfm(tokensAll)
dfmat_hes = dfm_select(dfmat_hes, pattern=hesitations)
tstat_freq_hes <- textstat_frequency(dfmat_hes, groups = d$group)

hesFreq = tapply(tstat_freq_hes$frequency,tstat_freq_hes$group,sum)
hesFreqPerMillion = (hesFreq / c(femaleTotal,maleTotal)) * 1000000

femaleDiff = (hesFreqPerMillion["female"] -
                 hesFreqPerMillion["male"]) /
                hesFreqPerMillion["male"]

llHes = logLikelihood.test(hesFreq["male"],
                   hesFreq["female"],
                   maleTotal,femaleTotal)
```

No significant difference in hesitation (-12%, G2 = 12.35, p = 0.002).

Balanced corpus

Re-create corpus with a balanced amount of lines from each game.

```
dM = data.frame(
  text = textMMini,
  group="male",stringsAsFactors = F
)
corpM = corpus(dM)
tokensM = tokens(corpM, remove_punct = TRUE)
maleTotal = sum(ntoken(tokensM))

dF = data.frame(
  text = textFMini,
  group="female",stringsAsFactors = F
)
corpF = corpus(dF)
tokensF = tokens(corpF, remove_punct = TRUE)
femaleTotal = sum(ntoken(tokensF))

d = rbind(dM,dF)
corpAll = corpus(d)
tokensAll = tokens(corpAll, remove_punct = TRUE)
```

Politeness

```
polMini = read.csv("../results/politenessMini.csv",stringsAsFactors = F)

politenessMini = NULL
for(feature in unique(polMini$feature)){
  nF = polMini[polMini$group=="female" & polMini$feature==feature,]$count
  nM = polMini[polMini$group=="male" & polMini$feature==feature,]$count
  propF = 1000000 * (nF/femaleTotal)
  propM = 1000000 * (nM/maleTotal)
  ll = logLikelihood.test(nM,nF,maleTotal, femaleTotal)
  politenessMini = rbind(politenessMini, data.frame(
    feature=feature, nFemale = nF,nMale = nM,
    nFemalePerMillionWords = propF,
    nMalePerMillionWords = propM,
    G2 = ll[1],
    p = ll[2]
  ))
}
politenessMini$p = p.adjust(politenessMini$p,method = "bonferroni")
```

Results:

```
knitr::kable(politenessMini, digits = 2)
```

feature	nFemale	nMale	nFemalePerMillionWords	nMalePerMillionWords	G2	p
Please	60	119	412.97	315.29	2.81	1.00
Please_start	126	154	867.24	408.02	37.59	0.00
HASHEDGE	2663	4746	18329.11	12574.58	232.68	0.00
Indirect_(btw)	11	22	75.71	58.29	0.49	1.00
Hedges	970	1584	6676.39	4196.83	123.98	0.00
Factuality	366	672	2519.13	1780.47	27.47	0.00
Deference	149	341	1025.55	903.48	1.64	1.00
Gratitude	319	480	2195.64	1271.77	54.49	0.00

feature	nFemale	nMale	nFemalePerMillionWords	nMalePerMillionWords	G2	p
Apologizing	299	443	2057.98	1173.73	53.66	0.00
1st_person_pl.	2504	5553	17234.73	14712.74	42.31	0.00
1st_person	3941	7774	27125.43	20597.31	191.89	0.00
1st_person_start	3669	6466	25253.29	17131.74	338.32	0.00
2nd_person	6091	12434	41923.63	32944.03	230.65	0.00
2nd_person_start	1465	3066	10083.42	8123.40	45.08	0.00
Indirect_(greeting)	199	655	1369.69	1735.43	8.91	0.24
Direct_question	1170	2723	8052.97	7214.62	9.74	0.16
Direct_start	1530	3043	10530.81	8062.46	70.35	0.00
HASPOSITIVE	6850	14241	47147.73	37731.70	223.32	0.00
HASNEGATIVE	5249	11280	36128.24	29886.49	125.78	0.00
SUBJUNCTIVE	60	132	412.97	349.74	1.12	1.00
INDICATIVE	85	143	585.04	378.88	9.64	0.17

The effects replicate:

- More hedging (+59%, G2 = 123.98, p < 0.001)
- More gratitude (+73%, G2 = 54.49, p < 0.001)
- More apologies (+75%, G2 = 53.66, p < 0.001)

Except for 'please':

- No significant difference for 'please' (+31%, G2 = 2.81, p = 1)

Swearing

```
dfmat_swears = dfm(tokensAll)
dfmat_swears = dfm_select(dfmat_swears, pattern=swears)
tstat_freq_swears <- textstat_frequency(dfmat_swears, groups = d$group)

swearFreq = tapply(tstat_freq_swears$frequency,tstat_freq_swears$group,sum)
swearFreqPerMillion = (swearFreq / c(femaleTotal,maleTotal)) * 1000000

femaleDiff = (swearFreqPerMillion["female"] -
                 swearFreqPerMillion["male"]) /
                swearFreqPerMillion["male"]

llSwear = logLikelihood.test(swearFreq["male"],
                   swearFreq["female"],
                   maleTotal,femaleTotal)
```

```
getStatText("swearing (mini)",femaleDiff, llSwear[1], llSwear[2])
```

```
## [1] "-46\\%, G2 = 46.77, p < 0.001"
```

Hesitation

```
dfmat_hes = dfm(tokensAll)
dfmat_hes = dfm_select(dfmat_hes, pattern=hesitations)
tstat_freq_hes <- textstat_frequency(dfmat_hes, groups = d$group)

hesFreq = tapply(tstat_freq_hes$frequency,tstat_freq_hes$group,sum)
hesFreqPerMillion = (hesFreq / c(femaleTotal,maleTotal)) * 1000000

femaleDiff = (hesFreqPerMillion["female"] -
                 hesFreqPerMillion["male"]) /
                hesFreqPerMillion["male"]
```

```r
llHes = logLikelihood.test(hesFreq["male"],
                   hesFreq["female"],
                   maleTotal,femaleTotal)
```

```r
getStatText("hesitations (mini)",femaleDiff, llHes[1], llHes[2])
```

```
## [1] "-12\\%, G2 = 1.6, p = 0.448"
```

References

Allan, K. & Burridge, K. 2006. Forbidden words: Taboo and the censoring of language. Cambridge: Cambridge University Press.

Baker, P 2014. Using Corpora to Analyze Gender. London: Bloomsbury Publishing

Bednarek, M., 2019. 'Don't say crap. Don't use swear words.'–Negotiating the use of swear/taboo words in the narrative mass media. Discourse, Context & Media, 29, p.100293.

Benoit K, Watanabe K, Wang H, Nulty P, Obeng A, Müller S, Matsuo A (2018). "quanteda: An R package for the quantitative analysis of textual data." *Journal of Open Source Software*, *3*(30), 774. doi: 10.21105/joss.00774

Brown, P., Levinson, S.C. and Levinson, S.C., 1987. Politeness: Some universals in language usage (Vol. 4). Cambridge university press.

Coates, J. 2003. Men Talk. United Kingdom: Blackwell Publishing Ltd.

Coates, Jennifer. 2004. Women, Men and Language: A Sociolinguistic Account of Gender Differences in Language. Edinburgh: Pearson.

Coyne, S.M., Callister, M., Stockdale, L.A., Nelson, D.A. and Wells, B.M., 2012. "A helluva read": profanity in adolescent literature. Mass Communication and Society, 15(3), pp.360-383.

Cressman, D.L., Callister, M., Robinson, T. and Near, C., 2009. Swearing in the cinema: An analysis of profanity in US teen-oriented movies, 1980–2006. Journal of Children and Media, 3(2), pp.117-135.

Danescu-Niculescu-Mizil, C., Sudhof, M., Jurafsky, D., Leskovec, J. and Potts, C., 2013. A computational approach to politeness with application to social factors. arXiv preprint arXiv:1306.6078.

DeFrank, M. and Kahlbaugh, P., 2019. Language choice matters: When profanity affects how people are judged. Journal of Language and Social Psychology, 38(1), pp.126-141.

Dixon, J.A. and Foster, D.H., 1997. Gender and hedging: From sex differences to situated practice. Journal of psycholinguistic research, 26(1), pp.89-107.

Dunning, Ted. (1993). Accurate Methods for the Statistics of Surprise and Coincidence. Computational Linguistics, Volume 19, number 1, pp. 61-74.

Gauthier, M. and Guille, A., 2017. Gender and age differences in swearing. Advances in swearing research: New languages and new contexts, pp.137-156.

Holmes, J., 1990. Hedges and boosters in women's and men's speech. Language & Communication, 10(3), pp.185-205.

Holmes, J., 2013. Women, men and politeness. Routledge.

Holtgraves, T. and Lasky, B. 1999. "Linguistic power and persuasion." Journal of Language and Social Psychology 18:2 (196-205).

Jan, S. and Rahman, M., 2020. Gender Determines Linguistic Features of One's Speech: Hedging and Interruptions in Male/Female Dialogues in One-Act Plays by male and female playwrights. Academic Journal of Social Sciences (AJSS), 4(4), pp.873-888.

Jay, T. and Janschewitz, K. (2008) The pragmatics of swearing. , Vol. 4 (Issue 2), pp. 267-288.

Jonathan P. Chang, Caleb Chiam, Liye Fu, Andrew Wang, Justine Zhang, Cristian Danescu-Niculescu-Mizil. 2020. "ConvoKit: A Toolkit for the Analysis of Conversations". Proceedings of SIGDIAL.

Karlsson Nordqvist, R., 2013. Gender Roles Via Hedging in Children's Films. Undergraduate thesis. https://www.diva-portal.org/smash/get/diva2:691798/FULLTEXT01.pdf

Knight, D., Adolphs, S. and Carter, R., 2013. Formality in digital discourse: a study of hedging in CANELC. In Yearbook of Corpus Linguistics and Pragmatics 2013 (pp. 131-152). Springer, Dordrecht.

Lakoff, R. (1973). Language and woman's place. Language in society, 2(1), 45-79.

McEnery, A. 2006. Swearing in English: Bad language, purity and power from 1586 to the present. London: Routledge

McEnery, A. and Xiao, Z., 2004. Swearing in modern British English: the case of fuck in the BNC. Language and Literature, 13(3), pp.235-268.

Mills, S., (2004) Class, gender and politeness. Multilingua 23:1-2, pp. 171-190.

Mirzapour, F., 2016. Gender differences in the use of hedges and first person pronouns in research articles of applied linguistics and chemistry. International Journal of Applied Linguistics and English Literature, 5(6), pp.166-173.

Mullany, L. (2004). Gender, politeness and institutional power roles: Humour as a tactic to gain compliance in workplace business meetings.

Nemati, A. and Bayer, J.M., 2007. Gender differences in the use of linguistic forms in the speech of men and women: A comparative study of Persian and English. Language in India, 7(9), pp.1-16.

Rayson P., Berridge D. and Francis B. (2004). Extending the Cochran rule for the comparison of word frequencies between corpora. In Volume II of Purnelle G., Fairon C., Dister A. (eds.) Le poids des mots: Proceedings of the 7th International Conference on Statistical analysis of textual data (JADT 2004), Louvain-la-Neuve, Belgium, March 10-12, 2004, Presses universitaires de Louvain, pp. 926 - 936.

Vold, E.T., 2006. Epistemic modality markers in research articles: a cross-linguistic and cross-disciplinary study. International Journal of Applied Linguistics, 16(1), pp.61-87.

Weisi, H. and Asakereh, A., 2021. Hedging devices in applied linguistics research papers: Do gender and nativeness matter?. Glottotheory, 12(1), pp.71-83.