Technical Appendix

0.1 Loading libraries

```r
library(harrypotter)
library(readtext)
library(quanteda)
library(R.utils)
library(cmu.textstat)
library(quanteda.textstats)
library(textdata)
library(SentimentAnalysis)
library(dplyr)
library(tidytext)
library(tidyr)
library(SentimentAnalysis)
library(stopwords)
library(knitr)
library(ggraph)
library(igraph)

devtools::install_github('wch/webshot') webshot::install_phantomjs()
# library(BiocManager)
BiocManager::install('https://bioconductor.org/biocLite.R')
source('https://bioconductor.org/biocLite.R') biocLite('EBImage')
library(pacman)
library(remotes)
library(magrittr)
library(BiocManager)
library(devtools)
pacman::p_load(devtools, knitr, magrittr, dplyr, ggplot2, rvest, sentimentSetsR,
    caret, textTinyR, text2vec, tm, tidytext, stringr, stringi, SnowballC, stopwords,
    wordcloud, prettydoc, cowplot, kable, utf8, corpus, glue, topicmodels, stm, wordcloud2,
    htmlwidgets, viridis)
```

1 Research Question 1

1.1 Loading all corpora

```r
Philosopher_Stone1 <- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/PhilosophersStone1997.t
Chamber_Secrets2<- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/ChamberofSecrets1998.txt")
Prisoner_Azkaban3<- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/PrisonerofAzkaban1999.tx
Goblet_Fire4<- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/GobletofFire2000.txt"))
```

```
Order_Phoenix5<- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/OrderofthePhoenix2003.txt")
HalfBlood_Prince6<- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/HalfBloodPrince2005.txt"
Deathly_Hallows7<- corpus(readtext("/Users/zhangxinge/Desktop/668/HarryPotter/DeathlyHallows2007.txt"))
```

1.2 Checking data

1.3 Tokenize all text and create DFM

```
stone_dfm <- Philosopher_Stone1 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()

Secrets_dfm <- Chamber_Secrets2 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()

Azkaban_dfm <- Prisoner_Azkaban3 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()

Fire_dfm <- Goblet_Fire4 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()

Phoenix_dfm <- Order_Phoenix5 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()

Prince_dfm <- HalfBlood_Prince6 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()

Hallows_dfm <- Deathly_Hallows7 %>%
  tokens(what="fastestword", remove_numbers=TRUE, remove_symbols = TRUE, remove_punct = TRUE) %>%
  dfm()
```

1.4 Create keyness tables using LL values

```
onetwo_kw <- keyness_table(Secrets_dfm, stone_dfm) %>% select(Token, Keyness=LL)
onetwo_kw1 <-onetwo_kw[1:10,]

twothree_kw <- keyness_table(Azkaban_dfm, Secrets_dfm) %>% select(Token, Keyness=LL)
twothree_kw1 <-twothree_kw[1:10,]

threefour_kw <- keyness_table(Fire_dfm, Azkaban_dfm) %>% select(Token, Keyness=LL)
threefour_kw1 <-threefour_kw[1:10,]

fourfive_kw <- keyness_table(Phoenix_dfm, Fire_dfm) %>% select(Token, Keyness=LL)
fourfive_kw1 <-fourfive_kw[1:10,]
```

```
fivesix_kw <- keyness_table(Prince_dfm, Phoenix_dfm) %>% select(Token, Keyness=LL)
fivesix_kw1 <-fivesix_kw[1:10,]

sixseven_kw <- keyness_table(Hallows_dfm, Prince_dfm) %>% select(Token, Keyness=LL)
sixseven_kw1 <-sixseven_kw[1:10,]

all <- cbind(onetwo_kw1, twothree_kw1, threefour_kw1, fourfive_kw1, fivesix_kw1, sixseven_kw1)
colnames(all) <- c("Token", "2-1 keyness","Token", "3-2 keyness","Token", "4-3 keyness","Token", "5-4 ke
all
```

2 Research Question 2

2.1 Creating token and dfm objects

```
stop_words <- c(stopwords(source = "smart"),
                stopwords(source = "snowball"),
                stopwords(source = "stopwords-iso"))

sorcerers_stone_tokens <- "harry_potter_corpus/sorcerers_stone.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

chamber_of_secrets_tokens <- "harry_potter_corpus/chamber_of_secrets.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

prisoners_of_azkaban_tokens <- "harry_potter_corpus/prisoner_of_azkaban.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

goblet_of_fire_tokens <- "harry_potter_corpus/goblet_of_fire.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

order_of_phoenix_tokens <- "harry_potter_corpus/order_of_phoenix.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

half_blood_price_tokens <- "harry_potter_corpus/half_blood_price.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

deathly_hallows_tokens <- "harry_potter_corpus/deathly_hallows.txt" %>%
  readtext() %>%
  corpus() %>%
  tokens(remove_punct = TRUE,
         remove_numbers = TRUE,
         remove_symbols = TRUE,
         what = "word") %>%
  tokens_tolower() %>%
  tokens_compound(pattern = phrase(multiword_expressions))

sorcerers_stone_dfm <- dfm(sorcerers_stone_tokens)
chamber_of_secrets_dfm <- dfm(chamber_of_secrets_tokens)
prisoners_of_azkaban_dfm <- dfm(prisoners_of_azkaban_tokens)
goblet_of_fire_dfm <- dfm(goblet_of_fire_tokens)
order_of_phoenix_dfm <- dfm(order_of_phoenix_tokens)
half_blood_price_dfm <- dfm(half_blood_price_tokens)
deathly_hallows_dfm <- dfm(deathly_hallows_tokens)
```

2.2 Computing collocate values for different characters

```
sorcerers_stone_MI <- collocates_by_MI(sorcerers_stone_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  add_row(token = "ginny", MI_1 = NA) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Sorcerer's Stone")

chamber_of_secrets_MI <- collocates_by_MI(chamber_of_secrets_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Chamber of Secrets")

prisoners_of_azkaban_MI <- collocates_by_MI(prisoners_of_azkaban_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Prisoner of Azkaban")

goblet_of_fire_MI <- collocates_by_MI(goblet_of_fire_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Goblet of Fire")

order_of_phoenix_MI <- collocates_by_MI(order_of_phoenix_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Order of the Phoenix")

half_blood_price_MI <- collocates_by_MI(half_blood_price_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Half Blood Prince")

deathly_hallows_MI <- collocates_by_MI(deathly_hallows_tokens, "harry") %>%
  filter(token %in% c("ron", "hermione", "dumbledore", "snape", "malfoy", "ginny")) %>%
  dplyr::select(token, MI_1) %>%
  rename(Character = token, Score = MI_1) %>%
  mutate(Book = "Deathly Hallows")
```

2.3 Creating data.frame with the relationship scores

```
harry_potter_MI <- rbind(sorcerers_stone_MI,
                         chamber_of_secrets_MI,
                         prisoners_of_azkaban_MI,
                         goblet_of_fire_MI,
                         order_of_phoenix_MI,
                         half_blood_price_MI,
                         deathly_hallows_MI)

books <- c("Sorcerer's Stone",
           "Chamber of Secrets",
           "Prisoner of Azkaban",
           "Goblet of Fire",
           "Order of the Phoenix",
           "Half Blood Prince",
           "Deathly Hallows")

harry_potter_MI$Character <- capitalize(harry_potter_MI$Character)

kable(harry_potter_MI)
```

2.4 Plotting graph

```
harry_relationship <- harry_potter_MI %>%
  ggplot(aes(x = factor(Book, level = books), y = Score, color = Character)) +
  geom_point(pch = 20) +
  geom_smooth(method = "gam", se = F) +
  geom_line(aes(group = Character)) +
  ylab("Relationship Score with Harry Potter") +
  xlab("") +
  ggtitle("Strength of relationships of different characters with Harry Potter across all seven books")
  theme(plot.title = element_text(size = 12), axis.text.x = element_text(angle = 45, hjust = 1))

ggsave("harry_potter_collocation_graph.png")
harry_relationship
```

3 Research Question 3

3.1 Collecting sentiment scores for different chapters across the different novels

```
sentiment_philosophers_stone <- analyzeSentiment(harrypotter::philosophers_stone)$SentimentGI
sentiment_chamber_of_secrets <- analyzeSentiment(harrypotter::chamber_of_secrets)$SentimentGI
sentiment_prisoner_of_azkaban <- analyzeSentiment(harrypotter::prisoner_of_azkaban)$SentimentGI
sentiment_goblet_of_fire <- analyzeSentiment(harrypotter::goblet_of_fire)$SentimentGI
sentiment_order_of_the_phoenix <- analyzeSentiment(harrypotter::order_of_the_phoenix)$SentimentGI
sentiment_half_blood_prince <- analyzeSentiment(harrypotter::half_blood_prince)$SentimentGI
sentiment_deathly_hallows <- analyzeSentiment(harrypotter::deathly_hallows)$SentimentGI
```

3.2 Creating a data frame object for the sentiment scores

```
sentiment_philosophers_stone_df <- data.frame(sentiment_scores = sentiment_philosophers_stone,
                                              Book = "Sorcerer's Stone")

sentiment_chamber_of_secrets_df <- data.frame(sentiment_scores = sentiment_chamber_of_secrets,
                                              Book = "Chamber Of Secrets")
```

```
sentiment_prisoner_of_azkaban_df <- data.frame(sentiment_scores = sentiment_prisoner_of_azkaban,
                                              Book = "Prisoner of Azkaban")

sentiment_goblet_of_fire_df <- data.frame(sentiment_scores = sentiment_goblet_of_fire,
                                              Book = "Goblet of Fire")

sentiment_order_of_the_phoenix_df <- data.frame(sentiment_scores = sentiment_order_of_the_phoenix,
                                              Book = "Order of the Phoenix")

sentiment_half_blood_prince_df <- data.frame(sentiment_scores = sentiment_half_blood_prince,
                                              Book = "Half Blood Prince")

sentiment_deathly_hallows_df <- data.frame(sentiment_scores = sentiment_deathly_hallows,
                                              Book = "Deathly Hallows")
```

3.3 Plotting a graph to compare the sentiment trend across the novels

```
sentiment_harry_potter <- rbind(sentiment_philosophers_stone_df,
                                sentiment_chamber_of_secrets_df,
                                sentiment_prisoner_of_azkaban_df,
                                sentiment_goblet_of_fire_df,
                                sentiment_order_of_the_phoenix_df,
                                sentiment_half_blood_prince_df,
                                sentiment_deathly_hallows_df)

kable(sentiment_harry_potter)

harry_potter_sentiment_analysis <- sentiment_harry_potter %>%
  ggplot(aes(x = seq(1, dim(sentiment_harry_potter)[1]), y = sentiment_scores)) +
  geom_point(pch = 20, aes(color = Book)) +
  geom_smooth(method = "gam", se = F, size = 0.75) +
  ylab("Sentiment Score") +
  xlab("Chapter Number") +
  ggtitle("Sentiment Trend Across All Harry Potter Books")

ggsave("harry_potter_sentiment_analysis.png")
harry_potter_sentiment_analysis
```