Roche_assignmnet

Siyan Chen

5/4/2019

PART 1:extract each article in the collection

```
df = readLines("/Users/siyanchen/Desktop/bm2_hw6/data/collection.txt")
```

```
## Warning in readLines("/Users/siyanchen/Desktop/bm2_hw6/data/
## collection.txt"): incomplete final line found on '/Users/siyanchen/Desktop/
## bm2_hw6/data/collection.txt'
```

```
# extract each article in the collection
section_index = grep(x = df, pattern = "^</doc>")
section_index
```

```
## [1] 182 237 445
```

```
section_1 = df[1:182]
section_2 = df[183:237]
section_3 = df[238:445]

page_s_index1 = grep(x = section_1, pattern = "^<p>") + 1
page_f_index1 = grep(x = section_1, pattern = "^</p>") - 1

page_s_index2 = grep(x = section_2, pattern = "^<p>") + 1
page_f_index2 = grep(x = section_2, pattern = "^</p>") - 1

page_s_index3 = grep(x = section_3, pattern = "^<p>") + 1
page_f_index3 = grep(x = section_3, pattern = "^</p>") - 1


### artical contents for artical 1, artical 2, artical 3
contents_1= vector("list", length = length(page_s_index1))
for (i in 1:length(page_s_index1)) {
  contents_1[[i]] = section_1[page_s_index1[i]:page_f_index1[i]]
  }

contents_2= vector("list", length = length(page_s_index2))
for (i in 1:length(page_s_index2)) {
  contents_2[[i]] = section_2[page_s_index2[i]:page_f_index2[i]]
}


contents_3= vector("list", length = length(page_s_index3))
for (i in 1:length(page_s_index3)) {
  contents_3[[i]] = section_3[page_s_index3[i]:page_f_index3[i]]
}


head(contents_1)
```

```
## [[1]]
## [1] "January 1, 1989, Sunday, Home Edition "
##
## [[2]]
## [1] "Book Review; Page 1; Book Review Desk "
##
## [[3]]
## [1] "1206 words "
##
## [[4]]
## [1] "NEW FALLOUT FROM CHERNOBYL; "
##
## [[5]]
## [1] "THE SOCIAL IMPACT OF THE CHERNOBYL DISASTER BY DAVID R. MARPLES (ST. MARTIN'S "
## [2] "PRESS: $35, CLOTH; $14.95, PAPER; 316 PP., ILLUSTRATED; 0-312-02432-0) "
##
## [[6]]
## [1] "By James E. Oberg , Oberg, a space engineer in Houston, is the author of "
## [2] "Uncovering Soviet Disasters: Exploring the Limits of Glasnost (Random House). "
```

```
head(contents_2)
```

```
## [[1]]
## [1] "January 1, 1989, Sunday, Home Edition "
##
## [[2]]
## [1] "Book Review; Page 10; Book Review Desk "
##
## [[3]]
## [1] "146 words "
##
## [[4]]
## [1] "CURRENT PAPERBACKS: WAITING FOR CHILDHOOD BY SUMNER LOCKE ELLIOTT (PERENNIAL "
## [2] "LIBRARY/ HARPER &amp;amp; ROW: $7.95) "
##
## [[5]]
## [1] "By ELENA BRUNET "
##
## [[6]]
## [1] "Set in Australia at the turn of the 20th Century, \"Waiting for Childhood\" is "
## [2] "the story of seven children left to cope for themselves after their parents "
## [3] "die. Their father, The Rev. William Lord, expires at the breakfast table one "
## [4] "morning. After the family leaves for a ramshackle house owned by a wealthy "
## [5] "cousin, the mother loses her mind and then her life in an accident. "
```

```
head(contents_3)
```

```
## [[1]]
## [1] "January 1, 1989, Sunday, Home Edition "
##
## [[2]]
## [1] "Business; Part 4; Page 3; Column 1; Financial Desk "
##
## [[3]]
## [1] "1299 words "
```

```
##
## [[4]]
## [1] "VIEWPOINTS; "
##
## [[5]]
## [1] "'89 WISH LIST: PROTECTION, TAXES AND PEACE; "
##
## [[6]]
## [1] "SOCIAL BENEFITS, DEFICIT REDUCTION ARE TOP PRIORITIES FOR THE NEW YEAR "
```

```
### string clean

article1 = as.data.frame(unlist(contents_1)) %>%
  rename(infor = c("unlist(contents_1)")) %>%
  mutate(infor = as.character(infor)) %>%
  mutate(infor = tolower(infor))


article2 = as.data.frame(unlist(contents_2)) %>%
  rename(infor = c("unlist(contents_2)")) %>%
  mutate(infor = as.character(infor)) %>%
  mutate(infor = tolower(infor))


article3 = as.data.frame(unlist(contents_3)) %>%
  rename(infor = c("unlist(contents_3)")) %>%
  mutate(infor = as.character(infor)) %>%
  mutate(infor = tolower(infor))
```

PART 1: Harsh Table

```
inspection_words1 =
  article1%>%
  unnest_tokens(word, infor)%>%
  count(word, sort = TRUE) %>%
  rename("n1" = c("n"))


inspection_words2 =
  article2%>%
  unnest_tokens(word, infor)%>%
  count(word, sort = TRUE)%>%
  rename("n2"= c("n"))


inspection_words3 =
  article3%>%
  unnest_tokens(word, infor)%>%
  count(word, sort = TRUE)%>%
  rename("n3"= c("n"))
```

```
### outer join and remove number


merged_df1 = merge(x = inspection_words1, y = inspection_words2, by = "word", all = TRUE)

merged_df_clean = merge(x = merged_df1, y = inspection_words3, by = "word", all = TRUE) %>%
  filter(!str_detect(word,"^[0-9]"))%>%
  gather(key = articles, value = count, n1:n3 ) %>%
  mutate(articles = recode(articles ,"n1" = "1", "n2" = "2", "n3" = "3")) %>%
  mutate(count = replace_na(count,0)) %>%
  spread(key = articles, value = count)


a1 = rep(1, nrow(merged_df_clean))
a2 = rep(2, nrow(merged_df_clean))
a3 = rep(3, nrow(merged_df_clean ))


merged_df  = merged_df_clean %>%
  cbind(a1, a2, a3)%>%
  unite(x1, c(a1, "1"),  sep = ",") %>%
  unite(y2, c(a2, "2"),  sep = ",") %>%
  unite(z3, c(a3, "3"),  sep = ",")

merged_df2= merged_df%>%
  unite(p, c(x1, y2, z3), sep = "] => [")

### separate hash table for each article
h1 = hashmap(key=merged_df$word, values = merged_df$x1)
h2 = hashmap(key=merged_df$word, values = merged_df$y2)
h3 = hashmap(key=merged_df$word, values = merged_df$z3)

### hash table(list 100)
h = hashmap(key=merged_df2$word, values = merged_df2$p)
options(hashmap.max.print = 20)
h
```

```
## ##   (character) => (character)
## ##   [aerospace] => [1,1] => [2,0] => [3,0]
## ##          [on] => [1,13] => [2,0] => [3,14]
## ##  [casualties] => [1,1] => [2,0] => [3,0]
## ##  [inspection] => [1,1] => [2,0] => [3,0]
## ##       [photo] => [1,1] => [2,0] => [3,0]
## ##  [elementary] => [1,0] => [2,0] => [3,2]
## ##        [skip] => [1,1] => [2,0] => [3,0]
## ##  [affections] => [1,0] => [2,1] => [3,0]
## ##         [rev] => [1,0] => [2,1] => [3,0]
## ##    [insights] => [1,2] => [2,0] => [3,1]
## ##    [discount] => [1,0] => [2,0] => [3,1]
## ##         [who] => [1,0] => [2,0] => [3,1]
## ##          [at] => [1,4] => [2,3] => [3,6]
## ##     [british] => [1,1] => [2,0] => [3,0]
## ##       [light] => [1,1] => [2,0] => [3,0]
```

```
## ##    [benefits] => [1,0] => [2,0] => [3,2]
## ##   [secondary] => [1,0] => [2,0] => [3,2]
## ## [journalists] => [1,1] => [2,0] => [3,0]
## ##       [never] => [1,1] => [2,0] => [3,0]
## ##         [one] => [1,4] => [2,1] => [3,2]
## ##         [...] => [...]
```

PART 2

```
### discrete distribution
total_inspection_word = merge(x = merged_df1, y = inspection_words3, by = "word", all = TRUE) %>%
  filter(!str_detect(word,"^[0-9]"))%>%
  gather(key = articles, value = count, n1:n3 ) %>%
  mutate(count = replace_na(count,0)) %>%
  spread(key = articles, value = count) %>%
  mutate(total_count = n1+n2+n3)

total_inspection_word %>%
  top_n(20) %>%
  ggplot(aes(x = total_count)) +
  geom_histogram()
```

```
## Selecting by total_count
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```