condensed_file

Julianna Alvord

2/14/2019

Loading streamed twitter data

loading dfs

```
#loading in full tweets df from sb_analysis_halves
load("/Users/juliannaalvord/Documents/nfl sentiment/sb_tweets_full.rda")

#df with 50 rows (includes the 50 starters of the 2019 sb)
starters <- read.csv("/Users/juliannaalvord/Documents/nfl sentiment/sb_starters.csv", stringsAsFactors =
```

creating list

```
#list of names
name <- starters$players

#cleaning twitter column, selecting that column, then filtering out those without twitter handle
twitter_clean <- starters %>%
  mutate(twitter_clean = sub("'", "", twitter)) %>%
  select(twitter_clean) %>%
  filter(!twitter_clean == "")

#list of twitter handles
twitter <- twitter_clean$twitter_clean

#full name and twitter handle for streaming
full_name <- c(name, twitter)

#making list for str_extract_all (92: 50 first names + 42 handles)
all_players <- paste(full_name, collapse='|')
```

searching tweet text for names and handles

```
#lowercasing player/handles for searching of text/quoted text
all_players_low = tolower(all_players)



#lower casing text and quoted text to be able to search without missing any players
full_more <- full %>%
  mutate(#pulling out the players from either text or quoted text
         name_text = str_extract_all(full_text_low, pattern = all_players_low),
         #hour created, just for vizs
         hour = hour(created_at))
```

unnesting df

```
#unnesting the name_text list column
full_more2 <- full_more %>%
  tidyr::unnest(name_text)
```

creating two dfs

```
#lowering twitter handles and player names for join
starters2 <- starters %>%
  mutate(twitter_clean = sub("'", "", twitter),
         twitter_clean2 = tolower(twitter_clean),
         name_clean = tolower(players)) %>%
  select(-c(players, twitter, twitter_clean))

#filtering for tweets that mention a player by their @
tweets_names <- full_more2 %>%
  filter(!grepl("@", name_text))

#filtering for tweets that mention a player by their full name
tweets_handles <- full_more2 %>%
  filter(grepl("@", name_text))
```

joining with starters data then row-binding

```
#tweets with names join
tweets_names2 <- tweets_names %>%
  left_join(starters2, by = c("name_text" = "name_clean"))

#tweets with handles join
tweets_handles2 <- tweets_handles %>%
  left_join(starters2, by = c("name_text" = "twitter_clean2"))

#row binding those two
tweets_final <- tweets_handles2 %>%
  bind_rows(tweets_names2) %>%
  #next code creates final name and twitter columns by filling in with name_text (what was joined on)
         #in tweets with names df, left join gets rid of "name_clean" col
  mutate(name_clean_final = ifelse(is.na(name_clean), name_text, name_clean),
         #in tweets with handles df, left join gets rid of "twitter_clean2" col
         twitter_clean_final = ifelse(is.na(twitter_clean2), name_text, twitter_clean2))
```

final cleaning

```
tweets_final <- tweets_final %>%
  #getting rid of incomplete names and twitter handles columns
  select(-c(name_clean, twitter_clean2))

#need to change one name
tweets_final <- tweets_final %>%
  mutate(name_clean_final = ifelse(name_clean_final %in% c("deatrich wise jr ", "deatrich wise jr,"),
                                   "deatrich wise jr.", name_clean_final),
         Race = ifelse(name_clean_final == "deatrich wise jr.", "black", Race))

#checking to make sure all the joins works

#number of tweets for each player
player_n <- tweets_final %>%
  group_by(name_clean_final) %>%
  summarise(n = n(),
            race = max(Race)) %>%
  arrange(desc(n))


kable(player_n %>% head(n = 20L), "latex", booktabs = T) %>%
  kable_styling(latex_options = "striped")
```

name_clean_final	n	race
tom brady	311496	white
julian edelman	77413	white
jared goff	58900	white
todd gurley	30833	black
stephen gostkowski	29453	white
aaron donald	20068	black
sony michel	19428	black
rob gronkowski	14361	white
andrew whitworth	12171	white
stephon gilmore	11023	black
johnny hekker	9638	white
kyle van noy	7398	black
patrick chung	6349	black
brandin cooks	5481	black
nickell robey-coleman	5039	black
marcus peters	4838	black
jason mccourty	4495	black
dont’a hightower	2856	black
chris hogan	2835	white
cordarrelle patterson	2003	black

sentiment analysis

adding sentiments to lexicon

```
#creating data frame with additional sentiments
extra<-data.frame(c("rings", "ring", "history", "clutch", "congrats", "dynasty", "goat", "g.o.a.t."),
               c("positive", "positive", "positive", "positive", "positive", "positive", "positive", "po
names(extra) <- c("word", "sentiment")

#binding to bing lexicon
bing_lex <- get_sentiments("bing")

sent_full <- rbind(bing_lex, extra)

#filtering out "patriot" since it should not have a sentiment for this analysis
sent_full <- sent_full %>%
  filter(!word == "patriot")
```

running loop to determine sentiments

```
#list of names for loop
names <- as.list(starters2$name_clean)

#empty list to add sentiments for each player
datalist = list()

for(i in 1:50) {

  #filter for each person in the starters df
  tweets <- tweets_final %>%
    filter(name_clean_final == names[i])

  #pick out words (each word is a row -- tidytext)
  words <- tweets %>%
    select(status_id, full_text_low) %>%
    unnest_tokens(word,full_text_low)

  #creating df of stop words
  my_stop_words <- stop_words %>%
    select(-lexicon) %>%
    bind_rows(data.frame(word = c("https", "t.co", "rt", "amp","4yig9gzh5t","fyy2ceydhi","78","fakenews

  #anti-join with stop words to filter those words out
  tweet_words <- words %>%
    anti_join(my_stop_words)

  #joining sentiments with non-stop words from tweets
  fn_sentiment <- tweet_words %>%
    left_join(sent_full)

  #creating df with n of sentiments
  df <- fn_sentiment %>%
    filter(!is.na(sentiment)) %>%
    group_by(sentiment) %>%
    summarise(n=n())

  #making df of sentiments for each person
  df_2 <- df %>%
  mutate(player = names[i]) %>%
  spread(key = sentiment, value = n)

  datalist[[i]] <- df_2



  ######uncomment next part if you want words df or sentiments dfs to be loaded into environment



  #creating name for dfs
  #df_name <- name

  #words_name <- paste(name, "word", sep = "_")

  #assigning df_name to df
  #assign(df_name, df_2)

}

#sentiments n for all players
sentiments_full_bing = do.call(rbind, datalist)
```

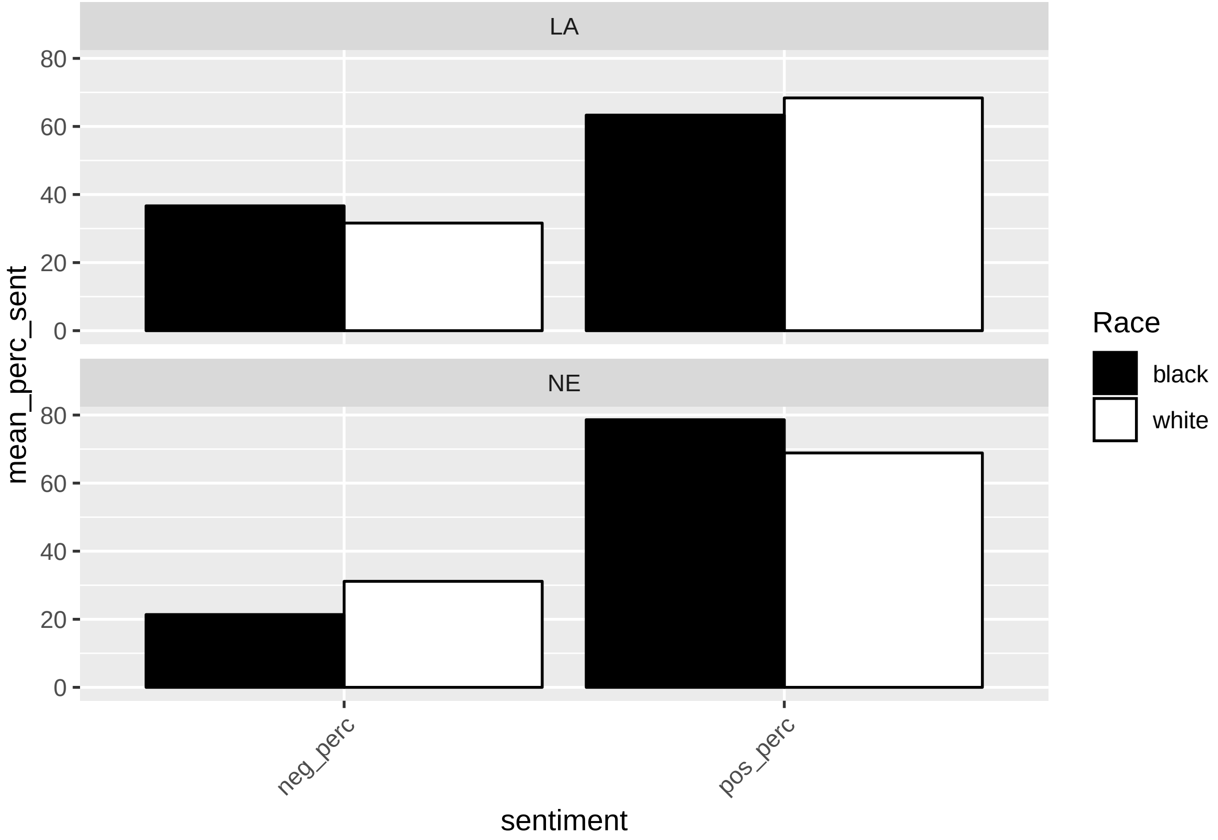
joining with starters data

```
#as.character to be able to join
sentiments_full_bing2 <- sentiments_full_bing %>%
  mutate(player = as.character(player))

#joining with starters and creating percentages
starters_sentiment2 <- starters2 %>%
  left_join(sentiments_full_bing2, by = c("name_clean" = "player")) %>%
  mutate(totalsentiment = positive+negative,
         neg_perc = negative/totalsentiment * 100,
         pos_perc = positive/totalsentiment *100)
```

grouping by team

```
#gathering by race, player, team, and sentiment
starters_sent_format <- starters_sentiment2 %>%
  select(name_clean, Race, team, position, 10:11) %>%
  gather(sentiment, n, 5:6)
```

```
#grouping by sentiment and race then making mean for each sentiment/race
starter_sent_2 <- starters_sent_format %>%
  dplyr::group_by(sentiment, Race, team) %>%
  summarise(mean_perc_sent = mean(n))

#same viz but by team as well
ggplot(starter_sent_2, aes(x = sentiment, y = mean_perc_sent, fill = Race)) +
  geom_bar(stat = "identity", position = "dodge", color = "black") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  scale_fill_manual(values=c("black", "white")) + facet_wrap(~team, ncol = 1)
```



```
#can remove pos_sent %  --> it's redundant
```