アメリカの銀行への口コミの分析

メンバー1〜5

2023-12-19

1. 研究のテーマと目的

　米国のトップ20のクレジットカード発行会社として認められている「メリック銀行」の口コミ状況を分析するのがこの研究の目的である。合計で約300万人のカード会員にサービスを提供し、約60憶ドルのクレジットを提供してきた。今回は19271個のデータをもとに分析。データにはメリック銀行に対する口コミ、その口コミに対するいわゆる「いいね」の数が書かれている。

1.1　研究の背景

　Rのテキストマイニングを通して、銀行評価のデータをもとにどの単語が一番多く書いてあるか、良い評価や悪い評価と結びつくものは何か探るのが目的。また、改善すべきことや優れているところを探すのも目的の一つ。そして、これからこの銀行がマーケティング戦略を実行するために役に立つことは何か、それを掘り出すことが求められている。

1.2 研究の問題意識

　二万行に近いコメントから顧客がどんな細かいサービスについて悪い評価良い評価をしたかについて探すのは大切だと思う。また、共通点という因子分析に近い分析も必要となってくる。ratingやlikeや地理など複数な項目を合わせて分析するのは精度が高いだと思う。

1.3 研究の目的

顧客に対する悪いことや良いことを探す。それに基づいてサービスの改善や維持など戦略を制定するのは不可欠な分析だ。

2. 研究の方法と手順

1. explored data analysis
2. sentiment analysis

3. データ分析と結果

3.1 探索的データ分析

3.1.1 パッケージロード

```
library(tidyverse)
```

```
## Warning: パッケージ 'tidyverse' はバージョン 4.2.3 の R の下で造られました
```

```
## Warning: パッケージ 'tibble' はバージョン 4.2.3 の R の下で造られました
```

```
## Warning: パッケージ 'dplyr' はバージョン 4.2.3 の R の下で造られました
```

```
## Warning: パッケージ 'lubridate' はバージョン 4.2.3 の R の下で造られました
```

```
## ── Attaching core tidyverse packages ──────────────────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ ggplot2   3.4.1     ✔ tibble    3.2.1
## ✔ lubridate 1.9.3     ✔ tidyr     1.3.0
## ✔ purrr     1.0.1
## ── Conflicts ────────────────────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the □]8;;http://conflicted.r-lib.org/□conflicted package□]8;;□ to force all conflicts to become errors
```

```
library(tidytext)
```

```
## Warning: パッケージ 'tidytext' はバージョン 4.2.3 の R の下で造られました
```

```
library(stringr)
library(DT)
```

```
## Warning: パッケージ 'DT' はバージョン 4.2.3 の R の下で造られました
```

```
library(igraph)
```

```
## Warning: パッケージ 'igraph' はバージョン 4.2.3 の R の下で造られました
```

```
##
##  次のパッケージを付け加えます: 'igraph'
##
##  以下のオブジェクトは 'package:lubridate' からマスクされています:
##
##     %--%, union
##
##  以下のオブジェクトは 'package:dplyr' からマスクされています:
##
##     as_data_frame, groups, union
##
##  以下のオブジェクトは 'package:purrr' からマスクされています:
##
##     compose, simplify
##
##  以下のオブジェクトは 'package:tidyr' からマスクされています:
##
##     crossing
##
##  以下のオブジェクトは 'package:tibble' からマスクされています:
##
##     as_data_frame
##
##  以下のオブジェクトは 'package:stats' からマスクされています:
##
##     decompose, spectrum
##
##  以下のオブジェクトは 'package:base' からマスクされています:
##
##     union
```

```
library(ggraph)
```

```
## Warning: パッケージ 'ggraph' はバージョン 4.2.3 の R の下で造られました
```

```
library(textdata)
```

```
## Warning: パッケージ 'textdata' はバージョン 4.2.3 の R の下で造られました
```

```
library(tm)
```

```
## Warning: パッケージ 'tm' はバージョン 4.2.3 の R の下で造られました
```

```
##  要求されたパッケージ NLP をロード中です
##
##  次のパッケージを付け加えます: 'NLP'
##
##  以下のオブジェクトは 'package:ggplot2' からマスクされています:
##
##      annotate
```

```
library(topicmodels)
```

```
## Warning: パッケージ 'topicmodels' はバージョン 4.2.3 の R の下で造られました
```

```
library(wordcloud)
```

```
## Warning: パッケージ 'wordcloud' はバージョン 4.2.3 の R の下で造られました
```

```
##  要求されたパッケージ RColorBrewer をロード中です
```

```
library(caret)
```

```
## Warning: パッケージ 'caret' はバージョン 4.2.3 の R の下で造られました
```

```
##  要求されたパッケージ lattice をロード中です
##
##  次のパッケージを付け加えます: 'caret'
##
##  以下のオブジェクトは 'package:purrr' からマスクされています:
##
##      lift
```

```
library(ggthemes)
```

```
## Warning: パッケージ 'ggthemes' はバージョン 4.2.3 の R の下で造られました
```

```
library(RColorBrewer)
fillColor = "#FFA07A"
fillColor2 = "#F1C40F"
```

3.1.2 前処理

```
bank<-read.csv("¥¥¥¥cckksv1/経営学部/専門ゼミ（鳥居弘志）/グループ研究（テキストマイニング）/A/Banks_sjis.csv")
head(is.na(bank))#欠損値かどうか確かめる
```

```
##      author  date location  bank  star  text  like
## [1,]  FALSE FALSE    FALSE FALSE FALSE FALSE  TRUE
## [2,]  FALSE FALSE    FALSE FALSE FALSE FALSE  TRUE
## [3,]  FALSE FALSE    FALSE FALSE FALSE FALSE FALSE
## [4,]  FALSE FALSE    FALSE FALSE FALSE FALSE FALSE
## [5,]  FALSE FALSE    FALSE FALSE FALSE FALSE FALSE
## [6,]  FALSE FALSE    FALSE FALSE FALSE FALSE FALSE
```

```
colSums(is.na(bank))#欠損値を合計する
```

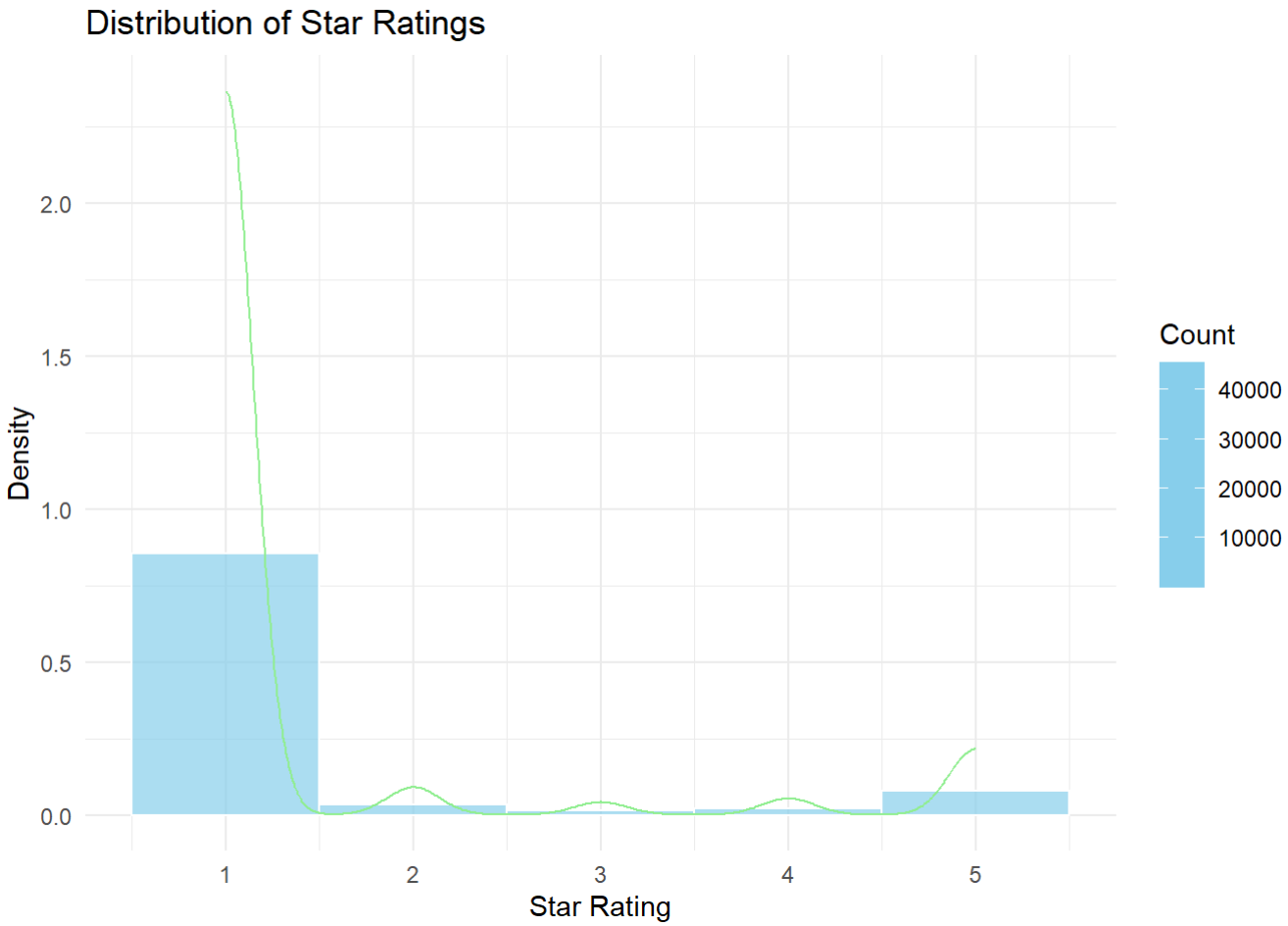
```
##   author     date location     bank     star     text     like
##        0        0        0        0        0        0     1502
```

```
# Drop rows with missing values in 'author' or 'text' columns
bank <- bank[complete.cases(bank[c('author', 'text')]), ]
# Replace NaN values in 'like' column with 0
bank$like[is.na(bank$like)] <- 0
```

3.1.3 評価の分布

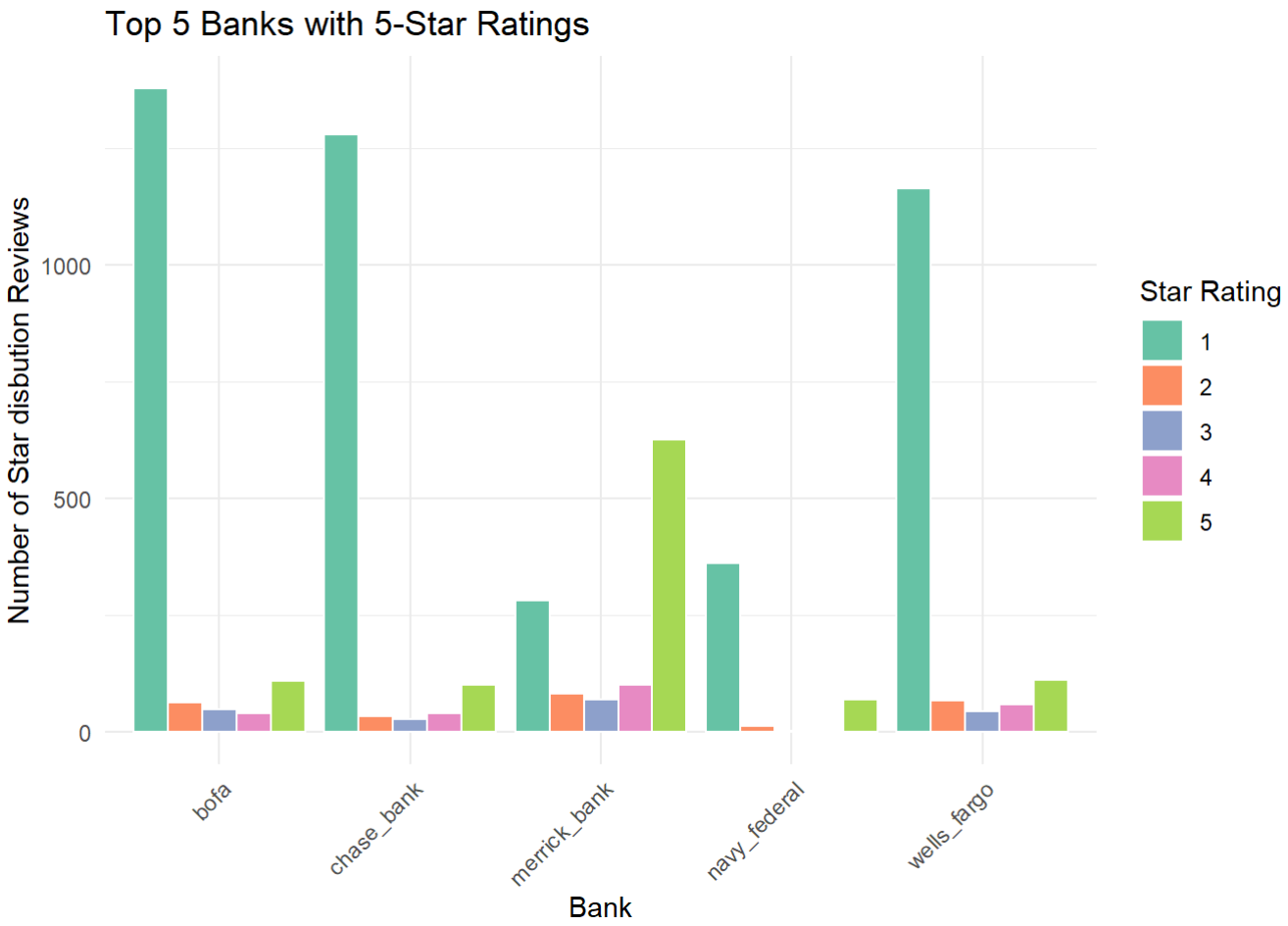
```
ggplot(bank, aes(x = star, y = ..density.., fill = ..count..)) +
  geom_histogram(binwidth = 1, color = "white", alpha = 0.7) +
  geom_density(alpha=0,color="lightgreen") +
  theme_minimal() +
  labs(title = "Distribution of Star Ratings",
       x = "Star Rating",
       y = "Density") +
  scale_fill_gradient("Count", low = "skyblue", high = "skyblue")
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## ℹ Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```


Distribution of Star Ratings
Density
2.0
1.5
1.0
0.5
0.0
1
2
3
4
5
Star Rating
Count
40000
30000
20000
10000

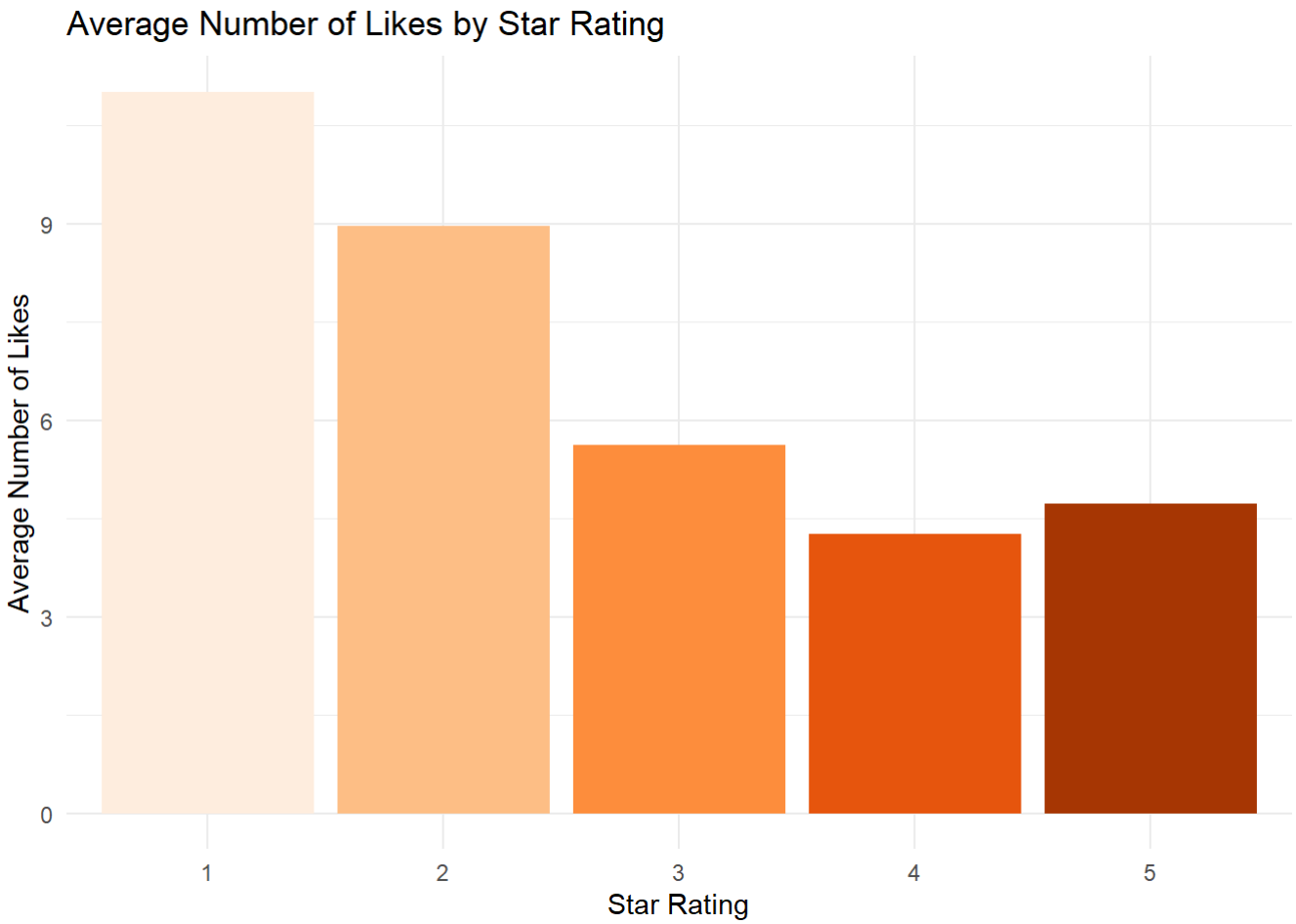
3.1.4 top5の銀行の評価分布

```
top_5_star_banks <- names(sort(table(bank$bank[bank$star == 5]), decreasing = TRUE)[1:5])
# Filter the data for the top 5 banks with 5-star ratings
filtered_data <- bank[bank$bank %in% top_5_star_banks, ]
color_palette <- brewer.pal(5, "Set2")
# Create a ggplot with a count plot for the top 5 banks with 5-star ratings
ggplot(filtered_data, aes(x = bank, fill = factor(star))) +
  geom_bar(position = "dodge", color = "white") +
  labs(title = "Top 5 Banks with 5-Star Ratings",
       x = "Bank",
       y = "Number of Star disbution Reviews") +
  scale_fill_manual(values = color_palette) +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  guides(fill = guide_legend(title = "Star Rating"))
```



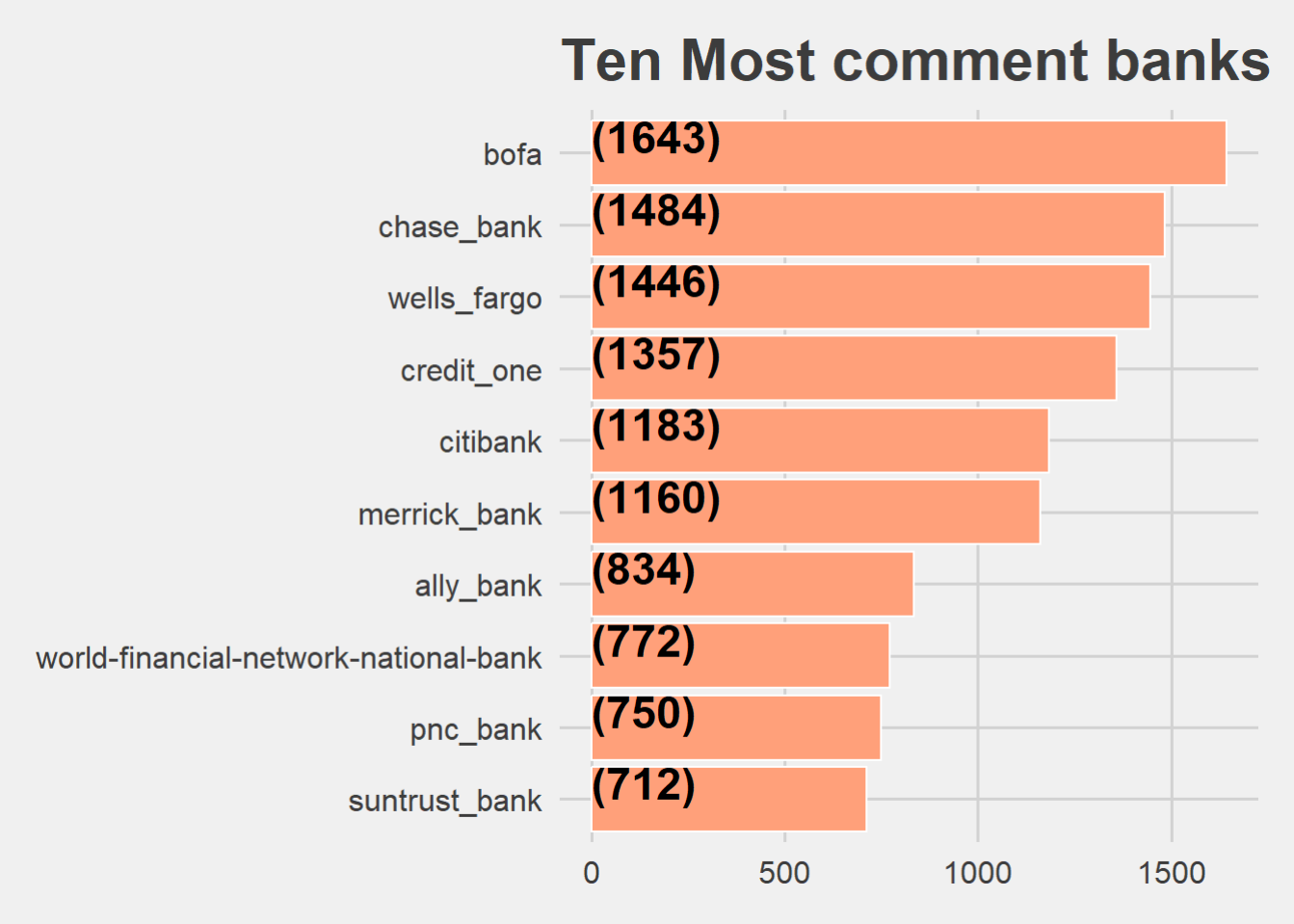
3.1.5 average likes by star rating

```
ggplot(bank, aes(x = factor(star), y = like)) +
  stat_summary(fun = "mean", geom = "bar", fill = brewer.pal(5,"Oranges"), position = "dodge") +
  labs(title = "Average Number of Likes by Star Rating",
       x = "Star Rating",
       y = "Average Number of Likes") +
  theme_minimal()
```


Average Number of Likes by Star Rating
Average Number of Likes
0
3
6
9
1
2
3
4
5
Star Rating

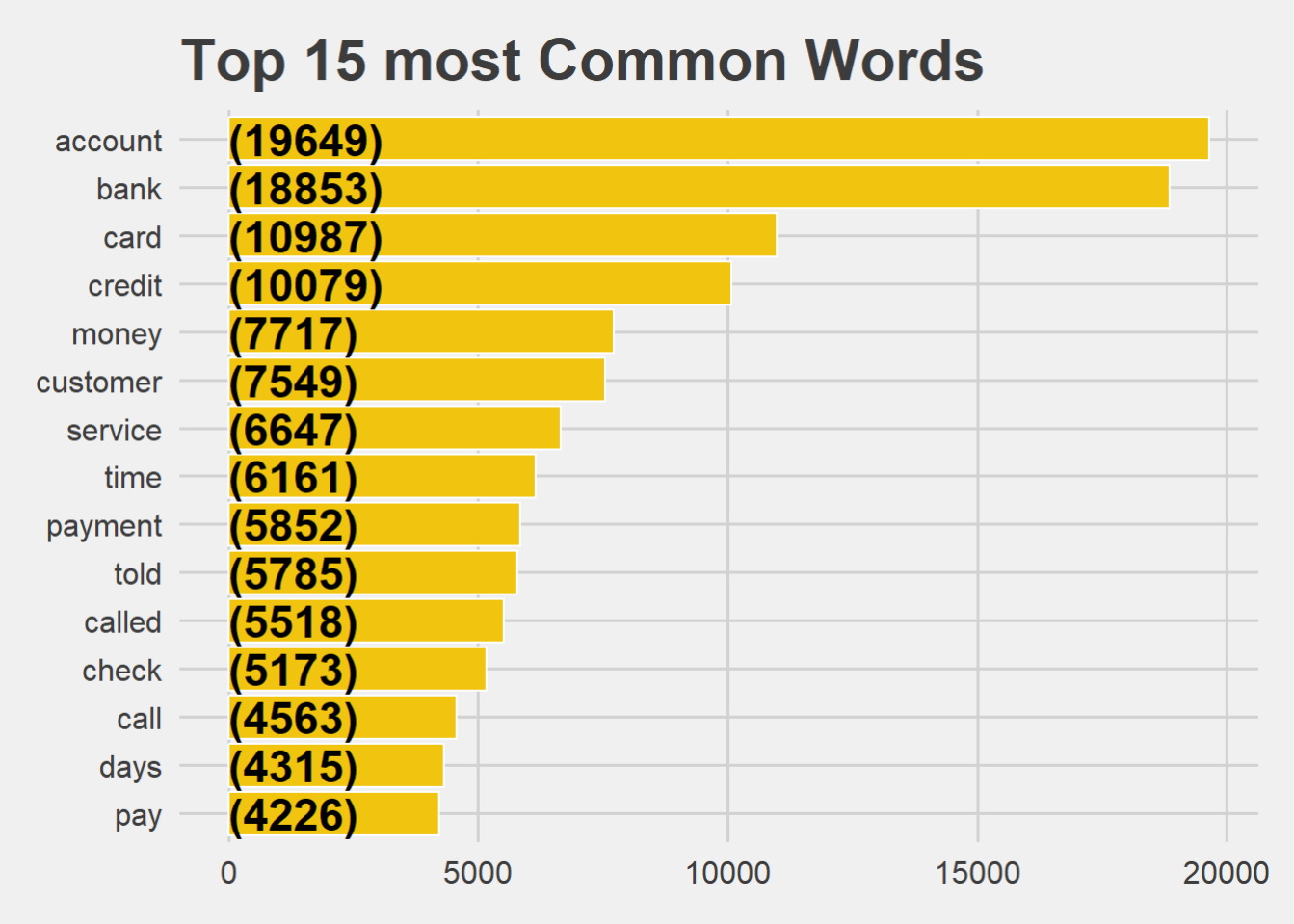
3.1.6 最もコメントされた銀行

```
topbank<-bank%>%
  group_by(bank)%>%
  tally(sort = TRUE)
ggplot(head(topbank,10),aes(x=reorder(bank,n),y=n))+
         geom_bar(stat = "identity",color="white",fill=fillColor)+
  geom_text(aes(x=bank,y=1,label=paste0("(",n,")",sep="")),
            hjust=0,vjust=0,size=6,colour="black",
            fontface="bold")+
  labs(x="name",y="Count Of Sentences",title="Ten Most comment banks")+
  coord_flip()+#転置
  theme_fivethirtyeight(base_size=15)
```


Ten Most comment banks
bofa
chase_bank
wells_fargo
credit_one
citibank
merrick_bank
ally_bank
world-financial-network-national-bank
pnc_bank
suntrust_bank
(1643)
(1484)
(1446)
(1357)
(1183)
(1160)
(834)
(772)
(750)
(712)
0
500
1000
1500

3.1.7 頻度が高い単語

```
bank%>%
  select(text)%>%
unnest_tokens(word,text)%>%#textデータを単語に分割し、word列として取得
  filter(!is.na(word))%>%#欠損値でない行をフィルタリング
  filter(!word %in% stop_words$word)%>%#情報量が低い単語を削除する
  count(word,sort=TRUE)%>%#単語出現回数を数えて、出現回数で降順に出す
  ungroup()%>%#データフレームをグループ化解除
  mutate(word = factor(word, levels = rev(unique(word))))%>%#factorしてユニークな単語を逆順にした順序でファクターレベルを設定
  head(15)%>%
ggplot(aes(x = word,y = n)) +
  geom_bar(stat='identity',colour="white", fill =fillColor2) +
  geom_text(aes(x = word, y = 1, label = paste0("(",n,")",sep="")),
  hjust=0, vjust=.5, size = 6, colour = 'black',
  fontface = 'bold') +
  labs(x = 'Word', y = 'Word Count',
  title = 'Top 15 most Common Words') +
  coord_flip() +
  theme_fivethirtyeight(base_size = 15)
```


Top 15 most Common Words
account
bank
card
credit
money
customer
service
time
payment
told
called
check
call
days
pay
(19649)
(18853)
(10987)
(10079)
(7717)
(7549)
(6647)
(6161)
(5852)
(5785)
(5518)
(5173)
(4563)
(4315)
(4226)
0
5000
10000
15000
20000

3.1.8　その頻度が高い単語を可視化

```
#wordcloudを作る
keyword<-bank%>%
  unnest_tokens(word,text)%>%
  filter(!is.na(word))%>%
  filter(!word %in% stop_words$word)%>%
  count(word,sort=TRUE)%>%
  ungroup()
keyword<-filter(keyword,!(word=="2"|word=="3"|word=="5"|word=="10"))#数字を削除
head(keyword,50)%>%
with(wordcloud(word, n, max.words = 50,colors=brewer.pal(9, "Paired")))#30サイズのwordcloudを作る
```


```
#part of speech ごとにworcloudにする
glimpse(parts_of_speech)#parts_of_speechデータセット
```

```
## Rows: 208,259
## Columns: 2
## $ word <chr> "3-d", "3-d", "4-f", "4-h'er", "4-h", "a'", "a-1", "a-axis", "a-b…
## $ pos  <chr> "Adjective", "Noun", "Noun", "Noun", "Adjective", "Adjective", "N…
```

```
unique(parts_of_speech$pos)#ユニックな値をとる
```

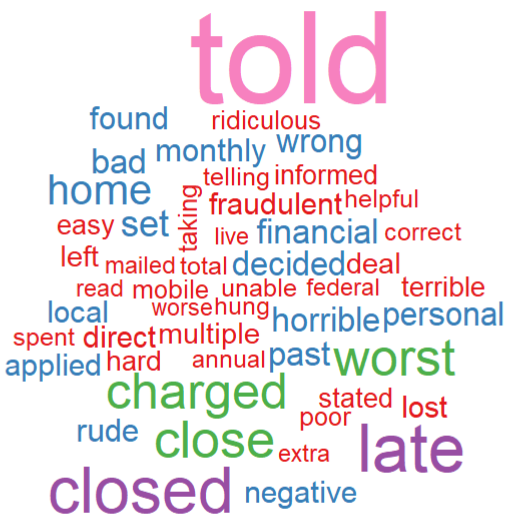
```
##  [1] "Adjective"             "Noun"                  NA
##  [4] "Plural"                "Adverb"                "Preposition"
##  [7] "Verb (transitive)"     "Verb (usu participle)" "Verb (intransitive)"
## [10] "Interjection"          "Noun Phrase"           "Conjunction"
## [13] "Definite Article"      "Pronoun"
```

3.1.9　その頻度が高い形容詞を可視化

```
#形容詞wordcloud
 bank%>%
  unnest_tokens(word, text) %>%
  filter(!word %in% stop_words$word) %>%
  left_join(parts_of_speech) %>%
  filter(pos == "Adjective") %>%
  count(word,sort = TRUE) %>%
  ungroup()  %>%
  head(50) %>%
   with(wordcloud(word, n, max.words =50,colors=brewer.pal(8, "Set1")))
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in left_join(., parts_of_speech): Detected an unexpected many-to-many relationship between `x` and `y`.
## ℹ Row 1 of `x` matches multiple rows in `y`.
## ℹ Row 103397 of `y` matches multiple rows in `x`.
## ℹ If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```


told
found
ridiculous
bad
monthly
wrong
telling
informed
home
taking
fraudulent
helpful
easy
set
live
financial
correct
left
mailed
total
decided
deal
read
mobile
unable
federal
terrible
local
worse
hung
horrible
personal
spent
direct
multiple
applied
hard
annual
past
worst
charged
stated
lost
poor
rude
close
extra
late
closed
negative

3.1.10　その頻度が高い動詞を可視化

```
# Verb(transitive)wordcloud
 bank%>%
   unnest_tokens(word, text) %>%
   filter(!word %in% stop_words$word) %>%
   left_join(parts_of_speech)%>%
     filter(pos =="Verb (transitive)") %>%
   count(word,sort = TRUE) %>%
   ungroup()  %>%
   head(50) %>%
   with(wordcloud(word, n, max.words =50,colors=brewer.pal(8, "Dark2")))
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in left_join(., parts_of_speech): Detected an unexpected many-to-many relationship between `x` and `y`.
## ℹ Row 1 of `x` matches multiple rows in `y`.
## ℹ Row 103397 of `y` matches multiple rows in `x`.
## ℹ If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

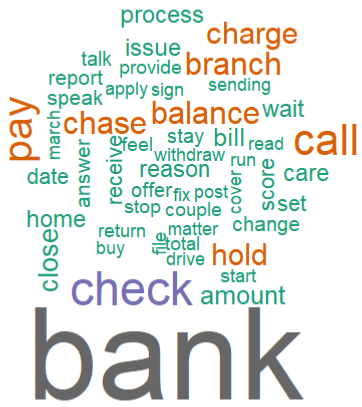
3.1.11　その頻度が高い動詞を可

視化

```
# Verb(intransitive)wordcloud
 bank%>%
   unnest_tokens(word, text) %>%
   filter(!word %in% stop_words$word) %>%
   left_join(parts_of_speech)%>%
    filter(pos =="Verb (intransitive)") %>%
   count(word,sort = TRUE) %>%
   ungroup()  %>%
   head(50)%>%
   with(wordcloud(word, n, max.words =50,colors=brewer.pal(8, "Dark2")))
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in left_join(., parts_of_speech): Detected an unexpected many-to-many relationship between `x` and `y`.
## ℹ Row 1 of `x` matches multiple rows in `y`.
## ℹ Row 103397 of `y` matches multiple rows in `x`.
## ℹ If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```



3.1.12　その頻度が高い名詞を可視化

```
# Noun wordcloud
 bank%>%
   unnest_tokens(word, text) %>%
   filter(!word %in% stop_words$word) %>%
   left_join(parts_of_speech)%>%
   filter(pos =="Noun") %>%
   count(word,sort = TRUE) %>%
   ungroup()  %>%
   head(50)%>%
   with(wordcloud(word, n, max.words =50,colors=brewer.pal(8, "Dark2")))
```

```
## Joining with `by = join_by(word)`
```

```
## Warning in left_join(., parts_of_speech): Detected an unexpected many-to-many relationship between `x` and `y`.
## ℹ Row 1 of `x` matches multiple rows in `y`.
## ℹ Row 103397 of `y` matches multiple rows in `x`.
## ℹ If a many-to-many relationship is expected, set `relationship =
##   "many-to-many"` to silence this warning.
```

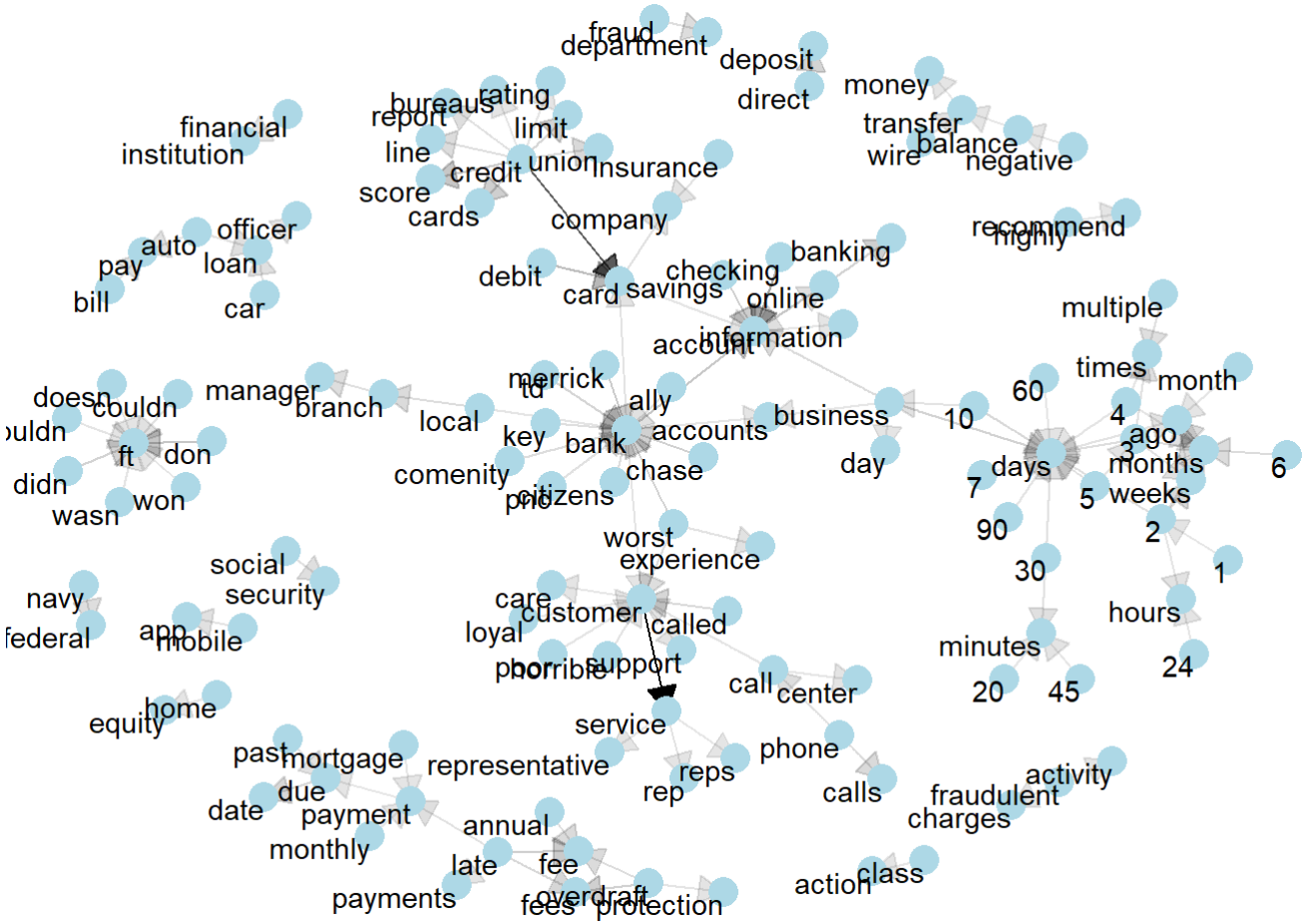

time
company
people
card
balance
branch
process
account
amount
day
fee
wait
transfer
cards
letter
charge
overdraft
close
bill
fargo
money
month
ft
information
loan
chase
send
due
home
fraud
payment
worst
cash
call
car
phone
service
pay
weeks
check
bank
hold
deposit
banking
person
mortgage
credit
customer
experience
business

3.1.13 共起ネットワーク図

```
count_bigrams <- function(dataset) {
  dataset %>%
  unnest_tokens(bigram, text, token = "ngrams", n = 2) %>%#1.テキストを単語やバイグラムにトークン化
  separate(bigram, c("word1", "word2"), sep = " ") %>% # 2. separate: バイグラムを2つの単語に分割
  filter(!word1 %in% stop_words$word,
         !word2 %in% stop_words$word) %>% # 3. stop_wordsを除外
    count(word1, word2, sort = TRUE) # 4. 単語のペアの出現回数を数える
}

visualize_bigrams <- function(bigrams) {
  set.seed(2016)
  a <- grid::arrow(type = "closed", length = unit(.15, "inches")) # 矢印の設定
  bigrams %>%
  graph_from_data_frame() %>%
    ggraph(layout = "fr") +# Force-directed layout (Kamada-Kawaiアルゴリズム) を使用
    geom_edge_link(aes(edge_alpha = n), show.legend = FALSE, arrow = a) + # エッジ (リンク) を描画
    geom_node_point(color = "lightblue", size = 5) + # ノード (頂点) を描画
    geom_node_text(aes(label = name), vjust = 1, hjust = 1) + # ノードのラベルを描画
    theme_void() # 背景を白にする
}

visualize_bigrams_individual <- function(bigrams) {  # 1. バイグラムの出現回数を数える
  set.seed(2016)
  a <- grid::arrow(type = "closed", length = unit(.15, "inches")) # 矢印の設定
  bigrams %>%
    graph_from_data_frame() %>%
    ggraph(layout = "fr") +
    geom_edge_link(aes(edge_alpha = n), show.legend = FALSE, arrow = a,end_cap = circle(.07, 'inches')) +
    geom_node_point(color = "lightblue", size = 5) +
    geom_node_text(aes(label = name), vjust = 1, hjust = 1) +
    theme_void()
}

bankWords <- bank %>%
  count_bigrams()  # バイグラムの出現回数を数える
```

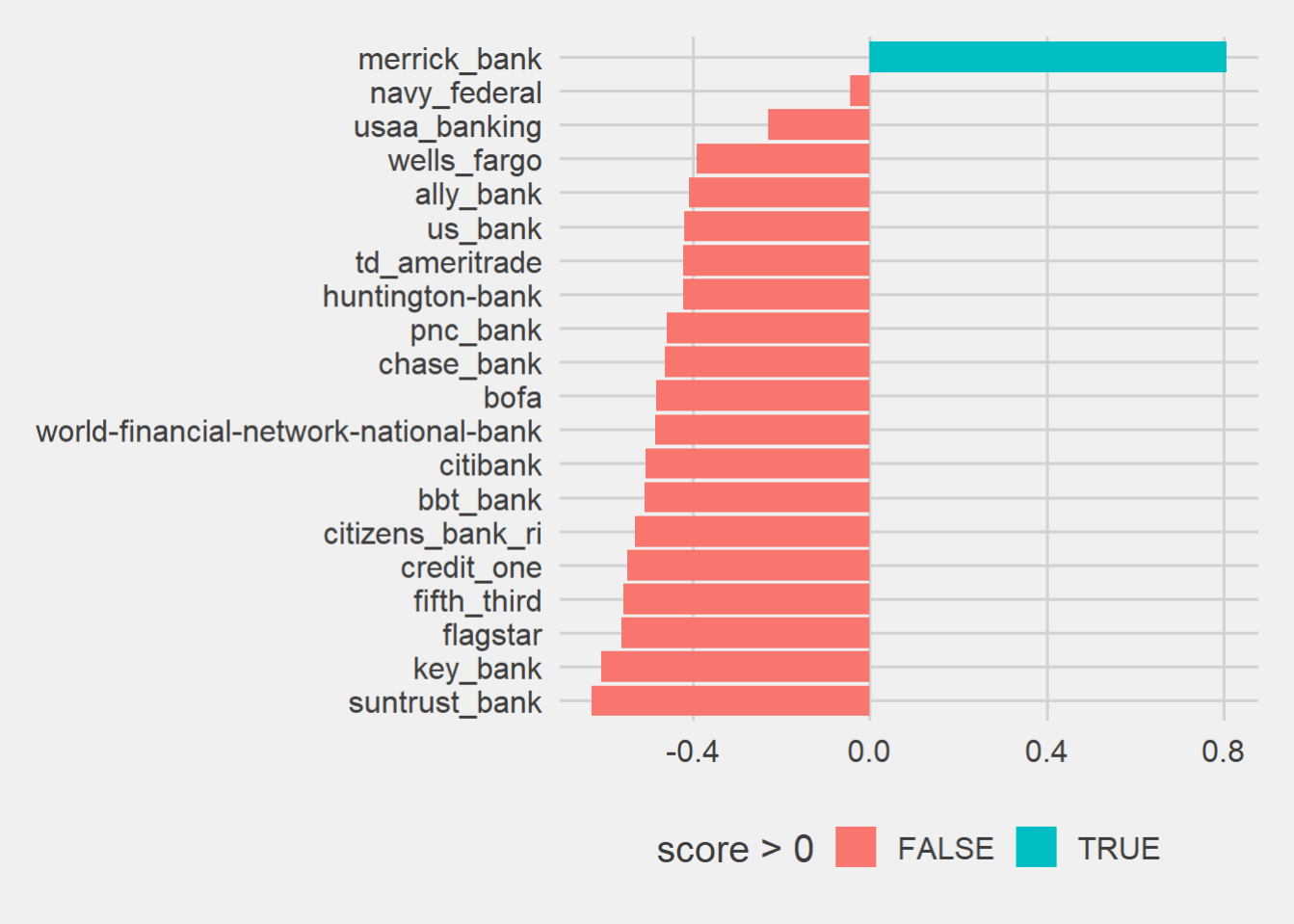
```
bankWords %>%
filter(n > 100) %>% # 50回以上出現したバイグラムを抽出
visualize_bigrams()
```

```
## Warning: Using the `size` aesthetic in this geom was deprecated in ggplot2 3.4.0.
## ℹ Please use `linewidth` in the `default_aes` field and elsewhere instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



3.1.14 sentiments 分析

```
visualize_sentiments <- function(bankWords) {
      bankWords_sentiments <- bankWords %>%
      inner_join(get_sentiments("afinn"), by = "word") %>% # AFINN辞書を用いて、単語の感情極性を計算
      group_by(bank) %>%
      summarize(score = sum(value * n) / sum(n)) %>% # バイグラムの感情スコアを計算
      arrange(desc(score)) # 感情スコアの高い順に並び替え
bankWords_sentiments %>%
    mutate(bank = reorder(bank, score)) %>% # 感情スコアの高い順に銀行名を並び替え
    ggplot(aes(bank, score, fill = score > 0)) + # 感情スコアが正かどうかによってバーを色分け
    geom_col(show.legend = TRUE) + #棒グラフを描画します。各バンクの感情スコアがバー
    coord_flip() + # 横向きに描画
    ylab("Average sentiment score") + theme_fivethirtyeight(base_size = 15) # ラベルを設定
}

 Top20bank = head(topbank,20)$bank #top20 bank

 bankWordsTop20bank <- bank %>%  #top20 bankのみのデータ
   unnest_tokens(word, text) %>%  # テキストを単語にトークン化
    filter(bank != "NA") %>%  # NAを除外
    filter( bank %in% Top20bank) %>% #top20 bankのみのデータ
    count(bank, word, sort = TRUE) %>% # 単語の出現回数を数える
    ungroup()

  visualize_sentiments(bankWordsTop20bank)   #top20 bankの感情分析の可視化
```



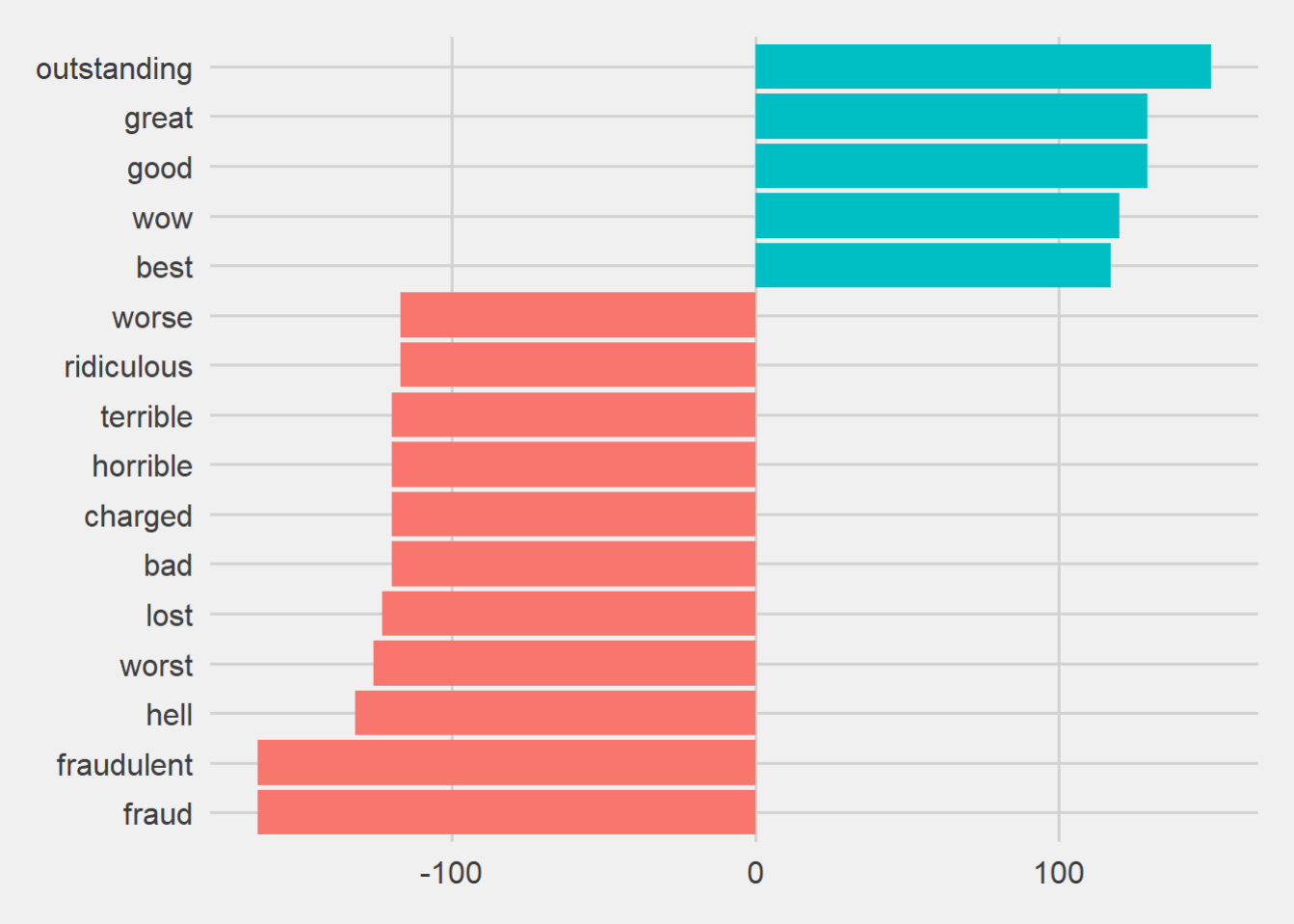

```
a<-aggregate(bank$star,by=list(bank$bank),FUN=mean) #銀行ごとの平均評価
arrange(a,desc(x)) #平均評価の高い順に並び替え
```

```
##                               Group.1        x
## 1                       bmo-harris-bank 5.000000
## 2                          merrick_bank 3.609483
## 3             arrowhead_credit_union_ca 3.000000
## 4       first-tech-federal-credit-union 3.000000
## 5                            sofi-money 1.813333
## 6                         discover-bank 1.780488
## 7                          navy_federal 1.672606
## 8                          usaa_banking 1.651332
## 9                           wells_fargo 1.539419
## 10                     pentagon_federal 1.475524
## 11                                 bofa 1.438831
## 12                           chase_bank 1.415094
## 13                    armed_forces_bank 1.377778
## 14                            ally_bank 1.353717
## 15                           associated 1.333333
## 16                              us_bank 1.323572
## 17                         td_ameritrade 1.302469
## 18                           arvest-bank 1.298246
## 19                      crescent_bank_la 1.254237
## 20                              bbt_bank 1.251244
## 21                         comerica_bank 1.242424
## 22                       huntington-bank 1.235897
## 23                      bank_of_the_west 1.234043
## 24                              pnc_bank 1.225333
## 25                         regions_bank 1.213058
## 26                       state_farm_bank 1.203704
## 27                         compass_bank 1.202020
## 28                         m_and_t_bank 1.195312
## 29                           royal_bank 1.187500
## 30                        suntrust_bank 1.185393
## 31                           credit_one 1.173176
## 32                          fifth_third 1.168142
## 33                     citizens_bank_ri 1.165138
## 34            first_national_bank_omaha 1.148649
## 35                           ing-direct 1.140078
## 36                             key_bank 1.130919
## 37                             flagstar 1.126551
```

```
## 38    presidents_choice_financial_canada 1.118182
## 39                            union_bank 1.114286
## 40                              citibank 1.114117
## 41                              tcf_bank 1.087209
## 42                                  hsbc 1.078704
## 43 world-financial-network-national-bank 1.069948
## 44                first_citizens_bank_nc 1.067568
## 45                         bofa_gift_card 1.000000
## 46                              cit-bank 1.000000
## 47          liberty-federal-credit-union 1.000000
```

3.1.15 positve and negative　分析

```
#positive and negative words analysis
 contributions <- bank %>%
   unnest_tokens(word, text) %>%
   filter(bank != "NA") %>%
   count(bank, word, sort = TRUE) %>%
   ungroup() %>%

   inner_join(get_sentiments("afinn"), by = "word") %>%
   group_by(word) %>%
   summarize(occurences = n(),
   contribution = sum(value))

 contributions %>%
   top_n(15, abs(contribution)) %>% #絶対値で上位15位を抽出
   mutate(word = reorder(word, contribution)) %>% # 感情スコアの高い順に単語を並び替え
   ggplot(aes(word, contribution, fill = contribution > 0)) + # 感情スコアが正かどうかによってバーを色分け
   geom_col(show.legend = FALSE) + # 棒グラフを描画
   coord_flip() +  # 横向きに描画
   theme_fivethirtyeight(base_size = 15)  # ラベルを設定
```


outstanding
great
good
wow
best
worse
ridiculous
terrible
horrible
charged
bad
lost
worst
hell
fraudulent
fraud
-100
0
100

3.1.18 top　positve and negative　comment

```
bank$id<-as.numeric(1:nrow(bank)) #idを追加
sentiment_lines  =  bank %>%
  unnest_tokens(word, text) %>%
  filter(bank != "NA") %>%
  inner_join(get_sentiments("afinn"), by = "word") %>%
  group_by(id,word) %>%
  summarize(sentiment = mean(value),
  words = n()) %>%
  ungroup() %>%
  filter(words >= 5)
```

```
## `summarise()` has grouped output by 'id'. You can override using the `.groups`
## argument.
```

```
result <- sentiment_lines %>%  #得点を昇順
  arrange(sentiment) %>%
  select(id)
 top5<-bank%>%
  filter(id %in% result$id) %>%  #対応するテキストを抽出
  select(id, text)
 head(top5)
```

```
##     id
## 1  321
## 2  749
## 3 1156
## 4 1397
## 5 1469
## 6 1574
##
text
## 1 I like the credit card and had no bad experience until I decided to pay my payment earlier than the due date. I decided to pay my bill on the 1st thinking I would get it over with so I wouldn't have to worry later. Well they ended up on the following month saying that I never paid. I called and was told that because I paid on the 1st it was put on the previous month as I paid 2 times for that month. I was so livid because I was never told in any of the paperwork that you would be penalized for paying early. Which the lady I was connected to also stated how it's not my fault because they don't tell anyone. But yet I still have to pay twice the payment amount they had me pay a payment and added another at the end of the month. Which makes it 2 payments I have to pay and was not told that during the conversation on the phone. I had to find out when I looked on my statement. Less on learned. I will not pay any of my credit card bills early again.
## 2
I had two fraud charges post to my account. When I noticed the charges on my August 2018 statement, I called to report the fraud. I was told the charges would be investigated, no payment would be required on the disputed charges, the charges would "suspended" pending the investigation, and a new card would be issued. Several weeks later I received my new card. My September 2018 statement cycled and the charges were still showing in the balance with a minimum payments requirement. I called back to Merrick Bank today to inquire why. The rep told me I was still responsible for the charges until such time as the investigation was completed. What a joke. The "rude service reps" are an even further stain on this company.
## 3 I have had my Merrick account for a few months. I got a call yesterday stating that there may have been fraudulent charges on my account. After talking about this with the representative, I discovered that the charges were in fact, fraudulent. They canceled my card and now they are freezing my account. I made one simple request. I wanted a statement so I could investigate the charges on my account. The website doesn't show pending charges. The rep was annoyed that I made this request. He literally stated, "Sir, I'm telling you what the charges are." Understandably, I want to see the charges for myself. I requested they mail me a statement of some kind showing all the charges on the account since the last time the horrible out of date website posted a charge. He acted as though I had asked for the moon and I was told I could not get that. After going back and forth with this guy for 10 minutes about why I wanted it, I decided to end the call.
## 4
I was trying to explain my issue and all of sudden he cut me of and literally screaming at me by saying STOP, STOP, STOP, STOP, STOP. I don't know how many times he said that but this kind of behavior it's not acceptable. And he cut my line off.
## 5
BB&T is a super bad bank. I have bad experience with them. 1) I recently called them to dispute fraud charge, their customer service never unable to reach a "REAL" person who can speak with you. 2) I have to get phone number of customer service thru thei
```

```
r health insurance department. (I contacted health insurance department before, I have their phone# in the email). 3) They refu
sed to talk fraud charge over the phone, need to go to branch. 4) I just got a call. They told me, "We cannot refund you for th
e fraud charge because time is over," I have been charged 450 fraud charge, that is the money I work hard to earn, why not able
to refund me? I never see such a bad bank, I have Bank of America. Chase bank, Discover bank cards, they ALL refund me if they
are fraud charge. So DO NOT USE BB&T anymore!!!!
## 6
I have been with BB&T for over a year now. I find myself walking away from them due to poor business practices. I left the bran
ch at 10625 Philips Hwy, Jacksonville, FL 32256 because of such poor business practices. On 02/12/2019 I deposited a check at t
his branches counter and was assured it would clear 02/21/2019 (TODAY). I clearly asked on the 12th what time-frame it would be
to clear this check. They explicitly responded 02/21/2019 (TODAY). As the reader might guess, it did not clear today. I walked
into this branch to inquire as to why it has not cleared. They could not give me a direct answer beyond, all we know is that it
is pending clear. We believe it will clear tonight at midnight. Once again, an indirect answer. I had them (BB&T) verify with t
he Check's Issuing Bank as to whether it had cleared on their end.
```

1 クレジットカードの支払いを早く払ったが、前月に計上され、2回分の支払いを請求された。契約書にはこのペナルティについての明記がなく、担当者も認めていたが、支払いを倍額でしないといけなくなった。

2 詐欺と訴えたのにも関わらず、担当者が適切な対応をしてくれなかったから怒ってる。

3 口座の不正請求を確認するために、明細書をお願いしたが、担当者に邪険に扱われた。

4 担当者に問題を説明しようとしたら、突然「やめろ、やめろ、やめろ」と叫ばれて 最終的に通話を切られました。

5 不正請求に対するカスタマーサービスが悪く、450ドルの不正請求に対する払い戻しも拒否された。

6 預けた小切手が約束された日にクリアされなかったことに対する対応が不十分で、 問い合わせても明確な回答が得られなかった。

3.1 分析について

今回使用している手法は最初から探索的データ分析をはじめ、主にテキストマイニングとして、sentiment やcloudwindやco-occurrance networkも使用していた。さらに、ベイジアンネットワークや他の機械学習手法は使用していない

3.2 分析結果の考察

結果として評価が低く、１星しかないコメントはほとんどである。ただしtop5の銀行のうち、merrick_bankは違い、５星の数は一番多い。そこからmerrick_bankのサービスが優れていることがわかる。また、コメントにtop15の単語を注目すると、account,bank,card,creditがtop4になった。そこから、コメントをする顧客はほとんど一般客だとわかる。さらに形容詞cloudwindを見ると、charged,late,toldなどの単語ができて,それ

らをほかのcloudwindを綱がいてクレジットカードがあるところ有料またはサービスが遅いなどのはマイナスとなる要因と考える。共起ネットワークからcreditとcardやterribeとexperienceなど単語が綱がいている。

4. まとめ

　まずは大体そのコメントの全体像を捕まえて顧客が気付いているところが分かった。ただ、raitngや星や地理や時間などのデータは使用されていなかったのは残念だ。またもっと高級な分析を使わなかったのは少し不足だと感じた。

参考文献

1. kaggle -Fun in Text Mining with Simpsons- Bukun - 2021 (https://www.kaggle.com/code/ambarish/fun-in-text-mining-with-simpsons/report)
2. kaggle-The Simpsons Dataset-Prashant Banerjee-2019 (https://www.kaggle.com/datasets/prashant111/the-simpsons-dataset)
3. kaggle-Customers_Reviews_on_Banks_EDA+NLP-Prajwal Dongre-2023 (https://www.kaggle.com/code/prajwaldongre/customers-reviews-on-banks-eda-nlp) }