TP3

Steven Chan

TP3

Introduction

À ce jour, le Parti conservateur du Canada agissait comme l'opposition officielle du Parti Libéral. Pendant plusieurs années le Parti conservateur et le Parti libéral se disputaient le gouvernement du Canada. Cependant, les conservateurs n'avaient pas connus du succès depuis 2011 avec le gouvernement de Stephen Harper. Ainsi, les Conservateurs ont du changé de chef à trois reprises. Cette recherche se concentraient sur les Conservateurs et leurs discours lors de assemblées du 27 au 29 mai 2019. La question de recherche était: quel étaient les enjeux les plus discutés par les députés conservateurs? Les données utilisées provennaient du site Lipad. Une platforme qui distribut des données textuelles concernant le Parlement canadien. Les données textuelles utilisées couvraient le 27, 28 et 29 mai 2019. Pour analyser les trois jours, une combinaison de trois banques de données avait eu lieu.

```
library(quanteda)
```

```
Package version: 3.3.1
Unicode version: 14.0
ICU version: 70.1

Parallel computing: 4 of 4 threads used.

See https://quanteda.io for tutorials and examples.
```

```
library(tidyverse)
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.2
--
```

```
v ggplot2 3.4.2     v purrr   1.0.1
v tibble  3.2.1     v dplyr   1.1.4
v tidyr   1.3.0     v stringr 1.5.1
v readr   2.1.3     v forcats 1.0.0
-- Conflicts ---------------------------------------- tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
```

```r
library(fs)
library(crayon)
```

```
Attaching package: 'crayon'

The following object is masked from 'package:ggplot2':

    %+%
```

```r
library(ggplot2)
```

L'analyse du dictionnaire utilisait la banque de donnée ci-dessous qui regroupaient plusieurs mots anglais.

```r
lexicoder <- dictionary(file = "policy_agendas_english.lcd",
                        format = "yoshikoder")
```

Cette ligne de code permettaient de créer une fonction qui permettait l'analyse du dictionnaire.

```r
run_dictionary <- function(data, text, dictionary) {
  tictoc::tic()
  if ( is.data.frame(data) != "TRUE") {
    stop(crayon::yellow('the argument "data" needs to be a dataframe'))
  }
  data <- data %>% dplyr::mutate(text = {{text}})
  if ( is.character(data$text) != "TRUE") {
    stop(crayon::yellow('The variable "text" needs to be a character vector'))
  }
  corpus <- quanteda::tokens(data$text)
  if ( quanteda::is.dictionary(dictionary) != "TRUE") {
```

```
    stop(crayon::yellow('Your "dictionary" needs to be in a dictionary format\n For more i
  }
  dfm    <- quanteda::dfm(quanteda::tokens_lookup(corpus, dictionary, nested_scope = "dict
  message(crayon::green("100% expressions/words found"))
  dataFinal   <- quanteda::convert(dfm, to = "data.frame")
  tictoc::toc()
  return(dataFinal)
}
```

Importation des données

En premier lieu, il fallait importé les trois bases de données qui regroupait les discours des trois jours. Ensuite, la fonction 'filter' permettait de conserver seulement les discours faites par les Conservateurs puisque la question de recherche concernait les Conservateurs. La fonction 'bind_rows' combinait les trois banques de données.

```
library(readr)
data_parl_1 <- read_csv("2019-5-28.csv")
```

```
Rows: 413 Columns: 15
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (12): hid, pid, speakeroldname, speakerposition, maintopic, subtopic, s...
dbl   (2): basepk, opid
date  (1): speechdate

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
data_28 <- data_parl_1 %>% filter(speakerparty == "Conservative")

data_27 <- read_csv("2019-5-27.csv")
```

```
Rows: 38 Columns: 15
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (12): hid, pid, speakeroldname, speakerposition, maintopic, subtopic, s...
dbl   (2): basepk, opid
date  (1): speechdate
```

```
i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
  data_27 <- data_27 %>% filter(speakerparty == "Conservative")

  data_parl_29 <- read_csv("2019-5-29.csv")
```

```
Rows: 263 Columns: 15
-- Column specification --------------------------------------------------------
Delimiter: ","
chr  (12): hid, pid, speakeroldname, speakerposition, maintopic, subtopic, s...
dbl   (2): basepk, opid
date  (1): speechdate

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
  data_29 <- data_parl_29 %>% filter(speakerparty == "Conservative")

  data_conser <- bind_rows(data_27,data_28, data_29)
```

Nettoyage de données

Le nettoyage des données consistaient à seulement 'select' tout les députés conservateurs ainsi que les discours qui venaient avec. Ensuite, il était important de transformer les discours en minuscule. Une étape qui facilitait l'analyse du dictionnaire qui était sensible au majuscule et au minuscule. Puis, la dernière étape était de retirer toutes les valeurs manquantes puisqu'elles n'étaient pas nécessaire à l'analyse.

```
  data_clean <- data_conser %>%
    select(speakerparty,speechtext) %>%
    mutate(speechtext = tolower(speechtext)) %>%
             na.omit()
```

L'analyse du dictionnaire était possible grâce à la fonction créer plus tôt. La fonction permettait d'identifier combien de fois certains mots revenait dans les catégories prédéterminées. Ainsi, le 'pivot_longer' permettait de regrouper toutes les variables sous une variable nommée "catégorie". Ainsi,les valeurs des catégories de mots étaient regroupé sous une variable 'n'. Le

'summarise' permettait d'additionner les valeurs identiques. Dans la base de donnée, il y avait plusieurs valeurs identiques qui avait un 'n' différent, puisqu'il y avait plusieurs discours de différentes journées. Ensuite, il fallait convertir les noms des variables en français.

```
data_object <- run_dictionary(data       = data_clean,
               text       = speechtext,
               dictionary = lexicoder) |>
  bind_cols(data_clean) |> select(-c(doc_id,speechtext)) %>%
  pivot_longer(!speakerparty, names_to = "categorie", values_to="n") |>
  ungroup() %>%  group_by(speakerparty,categorie)%>%
  filter(n>5) %>% summarise(n=sum(n)) %>%  mutate(
    categorie = case_when(categorie == "macroeconomics" ~ "Économie",
   categorie == "crime" ~ "Crime",
   categorie == "healthcare" ~ "Assurance maladie",
   categorie == "transportation" ~ "Transportation",
   categorie == "social_welfare" ~ "Sécurité sociale",
   categorie == "religion" ~ "Religion",
   categorie == "land-water-management" ~ "Contrôle des terres et eaux",
   categorie == "labour" ~ "Emplois",
   categorie == "foreign_trade" ~ "Exportation",
   categorie == "environment" ~ "Environement",
   categorie == "education" ~ "Éducation",
   categorie == "civil_rights" ~ "Droit civil",
    T ~ as.character(categorie)))%>%  na.omit()
```

```
100% expressions/words found
```

```
0.258 sec elapsed
```

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

Résultats

La question de recherche analysait les discours des députés conservateurs lors des assemblées du 27 au 29 mai 2019. L'analyse du dictionnaire démontrait que les conservateurs abordait le plus l'assurance maladie. Ensuite le droit civil, l'économie puis la religion. Les trois premier sujets n'étaient pas aussi surprenant puisque ses sujets représentaient des enjeux majeurs au Canada. Cependant la religion ressortait comme un sujet suprenant puisque la religion n'occupait tant de place dans la mise à l'agenda gouvernementale. L'assurance maladie se démarquait parmi les autres, un sujet qui pouvait être d'enjeu primordiale du 27 au 29 mai

2019. L'éducation représentait une surprise par rapport au bas nombre de mots utilisé à ce sujet.Les Conservateurs avaient mentionné peu de mots en lien avec l'éducation. Un résultat qui pouvait être expliqué par la séparation des compétences où l'éducation est une compétence provinciale.

```
data_object %>% ggplot(aes(x = n, y = categorie))+ geom_bar(
  stat= "identity", position = "dodge", na.rm = TRUE) + scale_x_continuous(
    breaks = seq (0,140, by = 20)) +
  labs( title = "Enjeux mentionné par le Parti conservateur du 27 au 29 mai",
    x = "Nombre de fois mentionné",
       y = "Enjeux") +
  theme_bw() + theme(panel.grid.major.x = element_blank(), title = element_text(size = 8),
        axis.text = element_text(size = 8, color = "black"))
```