

TP3-FAS 1001-Saffioti

Olivia Saffioti

Introduction

Selon la théorie de la construction sociale du genre, les électeurs évalueraient les politicien(ne)s en fonction des stéréotypes de genre. Si les politicien(ne)s n'adaptent par leur stratégie marketing aux stéréotypes, ils risqueraient un backlash, soit, des évaluations négatives de la part des électeurs (Rudman et Fairchild 2004; Coyle 2009, cités dans Grebelsky-Lichtman et Katz 2019). Ainsi les femmes et les hommes politiques, n'useraient pas des mêmes stratégies marketing afin de convaincre l'électorat. En effet, les femmes et les hommes répondraient souvent aux stéréotypes induits par le rôle attribué à chaque sexe dans la société. Ils conforteraient ainsi la vision des électeurs (Fox 1997). Cela incluerait l'image, les caractéristiques personnelles, les thèmes de campagne, et l'utilisation d'enjeux politiques spécifiques (Fox 1997). À titre d'exemple, les femmes politiques seraient perçues par l'électorat comme plus efficaces dans les domaines de l'éducation, de la santé, de l'environnement, des arts, de la protection des consommateurs, ou encore dans l'aide à apporter aux pauvres (Alexander and Andersen 1993; Koch 2000; McDermott 1998, cités dans Fox et Oxley 2003). À l'inverse, les hommes politiques seraient considérés comme plus compétents pour résoudre des crises militaires ou de police, des problèmes d'ordre économique, ou encore des enjeux liés au commerce. Ils seraient également identifiés comme plus légitimes pour s'occuper du contrôle de la criminalité ou encore de l'agriculture (Alexander and Andersen 1993; Koch 2000; McDermott 1998, cités dans Fox et Oxley 2003). En revanche, ces études ont surtout été menée au sein du contexte étasunien, et nous n'avons trouvé que peu de recherches qui visent les stratégies des femmes et des hommes politiques lors des débats électoraux. Il semble ainsi pertinent d'étudier si les femmes politiques insistent davantage sur les thématiques « féminines » (éducation, etc.) par rapport à leurs homologues masculins, dans un débat électoral qui n'est pas étasunien . C'est pourquoi, nous avons décidé d'étudier le débat électoral français opposant Marine Le Pen et Emmanuel Macron en 2017. Nous avons choisi ce débat car sa retranscription était facile d'accès. De plus, sélectionner un débat qui s'était produit après 2019 aurait biaisé nos résultats. En effet, la pandémie de Covid-19 demeurerait un sujet très discuté par les politiciens durant les campagnes suivant 2019. Ce contexte temporel ne nous aurait pas permis de vérifier si les femmes parlaient davantage de la santé que les hommes. À travers ce débat nous vérifierons si madame Lepen a plus insisté sur les enjeux de l'éducation, de la famille, de l'État Providence et de la santé que monsieur Macron. Cela nous permettra de vérifier si les postulats de la

théorie de la construction sociale du genre sont applicables à d'autres contextes. Pour ce faire, nous convertirons la retranscription pdf en une base de données. Puis, nous la nettoierons. Ensuite, nous élaborerons un dictionnaire que nous appliquerons à notre base de données. Afin de mieux visualiser et comprendre nos résultats, nous élaborerons un graphique.

Données et méthode

Afin de mener notre recherche, nous avons utilisé la base de données portant sur le débat présidentiel entre Marine Le Pen et Emmanuel Macron de 2017. Nous avons créé cette base de données à partir de la retranscription pdf du débat électoral (que nous avons trouvé sur le site internet "vie-publique.fr"). Nous avons nettoyé le texte en supprimant des caractères indésirables (qui ont remplacé des alinéas, etc.). En utilisant les fonctions "data.frame()", "mutate()", "str_squish()", "str_replace_all()", "str_extract_all()", "select()", "unnest()" et "slice()", nous avons cherché les noms des intervenants dans le débat et avons créé la variable `speaker_name` dans laquelle placer le nom des intervenants. Ensuite, nous avons sélectionné le texte et en avons fait une variable intitulée "text_data". Pour cela, nous avons mobilisé les fonctions "data.frame()", "mutate()", "str_squish()", "str_replace_all()", "str_split()", "unnest()", et "slice()". Nous avons fusionné les données des noms des intervenants (`speaker_names`) avec celles du texte afin de créer une nouvelle base de données. Dans cette nouvelle base de données, nous avons également créé une variable intitulée "année" pour y référer l'année du débat, une variable nommée "pays" pour y référer le pays où le débat a lieu, une variable intitulée "id" pour savoir à quel tour des élections le débat correspond, une variable nommée "numéro d'intervention" pour mettre en lumière l'ordre auquel les intervenants parlent, et une variable intitulée "parti" afin de référer le parti d'appartenance ou la profession des intervenants du débat. Pour ce faire, nous avons utilisé les fonctions "bind_cols()", "mutate()", "str_replace_all()", "str_squish()", "case_when()" et "select()". Nous avons finalement enregistré notre nouvelle base de données dans notre ordinateur à l'aide de la fonction "write_csv()".

Ensuite, nous avons importé cette base de données sur Rstudio afin de la nettoyer et d'opérer une analyse de dictionnaire. Dans un premier temps, nous avons créé notre dictionnaire afin de pouvoir analyser la fréquence de certains mots dans le discours des candidats. Personnaliser notre dictionnaire nous a permis de faire une analyse qui corrélait avec le contexte (les thématiques évoquées dans le débat) et notre sujet de recherche. Cela nous a permis de prendre en compte des mots spécifiques tels que "protection sociale" ou "quotient". Les mots du dictionnaire sont reliés aux thématiques "féminines" de l'éducation, de la santé, de la famille, et de l'État providence (lutte contre la pauvreté, etc.). Afin de créer le dictionnaire, nous avons mobilisé la fonction "list()", et la fonction "dictionnary()".

Puis, nous avons nettoyé notre base de données : nous avons retiré les ponctuations du texte et converti le texte en minuscules. Cela nous permettait par la suite de faciliter notre analyse de dictionnaire (un mot présent dans notre dictionnaire en minuscules ne serait pas relevé par notre analyse de dictionnaire dans le texte s'il comprenait des majuscules par exemple). Pour

cela, nous avons mobilisé les fonctions “tolower()” et “gsub()”. Nous avons disposé notre texte nettoyé dans une nouvelle colonne nommée “text_clean”. Ensuite, nous avons sélectionné les variables que nous voulions conserver dans notre base de données, afin de retirer les variables inutiles. Nous avons décidé de conserver les variables “party”, “text_clean” et “speaker”. Dans la variable “speaker” nous avons conservé uniquement deux speakers (deux catégories) : “Le Pen” et “Macron”. Nous avons choisi de les conserver car nous nous centrons uniquement sur ces deux candidats dans le cadre de notre étude. Pour cela, nous avons utilisé les fonctions “select()” et “filter()”. Ensuite, nous avons procédé à notre analyse de dictionnaire, et transféré les résultats dans une nouvelle base de données intitulée “debat_macron_lepen_resultat”. Pour ce faire, nous avons mobilisé la fonction “run_dictionary()”, la fonction “bind_cols()”, et la fonction “select()”. “run_dictionary()” nous a permis de “rouler” notre dictionnaire afin qu’il relève les mots dans le texte. La fonction bind_cols a permis de conserver les colonnes présentes dans la base de données “debat_macron_lepen”, dans notre nouvelle base de données “debat_macron_lepen_resultat”. La fonction select() nous a permis de sélectionner les colonnes de la base de données “données debat_macron_lepen” que nous ne voulions pas garder dans “debat_macron_lepen_resultat” (ici, nous n’avons pas conservé la variable “party”, et la variable “doc_id”).

Résultats

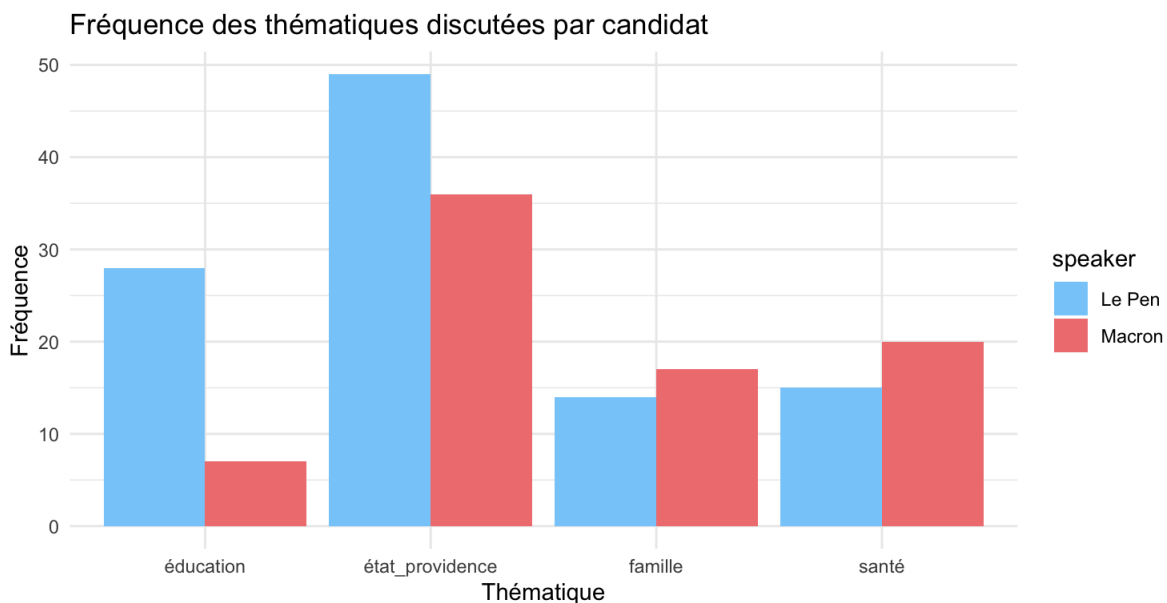


Figure 1: Graphique

Les résultats que nous avons obtenus indiquent que Marine Le Pen et Emmanuel Macron n’auraient pas insisté de la même manière sur les enjeux de la santé, de la famille, de l’État

providence et de l'éducation. En effet, 49 mots liés à l'État providence ont été relevés dans le discours de Marine Le Pen. Comparativement, 36 mots ont été relevés dans le discours d'Emmanuel Macron. 13 mots de plus ont été relevés pour Marine Le Pen par rapport à Emmanuel Macron. Bien que les deux candidats aient beaucoup insisté sur cette thématique, il semblerait que madame Le Pen ait plus insisté sur les éléments liés à l'état-providence que monsieur Macron. De plus, 28 mots liés à l'éducation ont été soulevés dans les paroles de Le Pen, contre seulement 7 pour Macron. Il y a donc un écart important de 21 mots entre Le Pen et Macron. Madame Le Pen aurait donc plus insisté sur la thématique de l'éducation que monsieur Macron. En revanche, concernant la thématique de la famille, l'écart entre les deux candidats n'est pas élevé : il est seulement de 3 mots. En effet, 14 mots ont été relevés dans le discours de madame Le Pen contre 17 pour monsieur Macron. Il semblerait donc que les deux candidats aient insisté de manière à peu près équivalente sur la thématique de la famille. Enfin, concernant la thématique de la santé, 15 mots ont été relevés dans le cas de madame Le Pen et 20 dans le cas de monsieur Macron. Là encore, l'écart n'est pas important entre Le Pen et Macron (il est seulement de 5 mots). Il semblerait que monsieur Macron et madame Lepen aient insisté de manière à peu près équivalente sur la thématique de la santé.

Dans l'ensemble, plus de mots liés à la famille et à la santé ont été relevés pour Emmanuel Macron. En revanche, l'écart avec madame Le Pen n'est pas élevé. Les écarts de mots pour ces deux thématiques ne sont pas significatifs : ils ne démontrent pas que Macron aurait plus insisté que Lepen sur ces deux thématiques. En revanche, l'écart entre Macron et Lepen est plus prononcé concernant la thématique de l'éducation et de l'état providence. Par conséquent, Marine Le Pen semble, généralement, avoir davantage insisté sur les thématiques "féminines" par rapport à Emmanuel Macron. Donc, il semblerait que, dans le cas du débat présidentiel entre Emmanuel Macron et Marine Lepen de 2017, le postulat selon lequel les femmes politiques insisteraient davantage sur les thématiques "féminines" (santé, éducation, etc.) est validé.

Conclusion

En somme, nous avons étudié le débat présidentiel de 2017 opposant Emmanuel Macron à Marine Lepen afin de vérifier si madame Lepen insistait davantage sur les thématiques "féminines" que monsieur Macron. Les résultats issus de notre analyse de dictionnaire ont montré que madame Lepen insistait davantage sur les thématiques de l'État providence et de l'éducation. De plus, Macron et Lepen insistaient avec une fréquence de mots quasi-identique sur la famille et la santé. Nous avons conclu que, généralement, Marine Lepen avait plus insisté sur les thématiques féminines qu'Emmanuel Macron dans le cadre de ce débat électoral. En revanche, notre recherche a une faible validité externe : il aurait fallu étudier plusieurs débats électoraux de Marine Lepen pour savoir si la candidate a insisté sur ces thématiques dans le cadre d'autres débats électoraux. De surcroît, il aurait été pertinent d'étudier des débats électoraux incluant d'autres femmes issues de partis politiques différents. Cela aurait permis de contrôler pour le parti d'appartenance, et donc de vérifier si l'insistance sur les enjeux féminins dépend du parti politique des politicien(ne)s (et non du genre).

Bibliographie

Fox, Richard Logan. 1997. *Gender Dynamics in Congressional Elections*. Thousand Oaks: Éditions SAGE. <https://books.google.ca/books?hl=en&lr=&id=x2w1BfJwRkAC&oi=fnd&pg=PR13&dq=gender+w>

Fox, Richard L., et Zoe M. Oxley. 2003. “Gender Stereotyping in State Executive Elections: Candidate Selection and Success”. *The Journal of Politics* 65 (3): 833-50. <https://doi.org/10.1111/1468-2508.00214>.

Grebelsky-Lichtman, Tsira, Roy Katz. 2019. ” When a man debates a woman : Trump vs.Clinton in the first mixed gender presidential debate “. *Revue Journal of Gender Studies* 28 (6): 699-719. <https://www.tandfonline.com/doi/full/10.1080/09589236.2019.1566890>

Annexe

```
# Convertir le texte pdf en base de données csv :
```

```
## Libraries
```

```
library(pdftools)
```

Using poppler version 23.04.0

```
library(tidyverse)
```

```
-- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
```

```
v dplyr      1.1.4      v readr      2.1.5
v forcats    1.0.0      v stringr    1.5.1
v ggplot2    3.4.4      v tibble     3.2.1
v lubridate  1.9.3      v tidyr      1.3.0
v purrr      1.0.2
```

```
-- Conflicts ----- tidyverse_conflicts() --
```

```
x dplyr::filter() masks stats::filter()
```

```
x dplyr::lag()     masks stats::lag()
```

```
i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become
```

```
library(lubridate)
```

```

## Data

text_1 <- suppressMessages(pdftools::pdf_text("/Users/oliviasaffioti/Desktop/fas_1001_Saffiotti.pdf"))

## Settings

split_vector <- paste0(c("Mme Le Pen :", "Mme Saint-Cricq :", "M. Jakubyszyn :", "M. Macron :"))

## Nettoyage de code

## Chercher les nom des personnes concernées dans le débat

speaker_names <- data.frame(text = paste0(text_1, collapse = " ")) |>
  mutate(text = str_squish(str_replace_all(text, c("[:punct:]C2[:punct:]AB" = "",
                                                    "[:punct:]C2[:punct:]BB" = ""))),
         text = str_replace_all(text, c("Emmanuel Macron :" = "M. Macron :",
                                         "Marine Le Pen :" = "Mme Le Pen :",
                                         "Christophe Jakubyszyn :" = "M. Jakubyszyn :",
                                         "Nathalie Saint-Cricq :" = "Mme Saint-Cricq :")),
         speaker = str_extract_all(text, split_vector)) |>
  select(-text) |>
  unnest(speaker) |>
  slice(-805)

## Données de prises de parole (text data)

text_data <- data.frame(text = paste0(text_1, collapse = " ")) |>
  mutate(text = str_squish(str_replace_all(text, c("[:punct:]C2[:punct:]AB" = "",
                                                    "[:punct:]C2[:punct:]BB" = ""))),
         text = str_replace_all(text, c("Emmanuel Macron :" = "M. Macron :",
                                         "Marine Le Pen :" = "Mme Le Pen :",
                                         "Christophe Jakubyszyn :" = "M. Jakubyszyn :",
                                         "Nathalie Saint-Cricq :" = "Mme Saint-Cricq :")),
         text = str_split(text, split_vector)) |>
  unnest(text) |>
  slice(-c(1,806))

## Fusion des deux variables clés et création de la base de données finale

Data_debate_2017 <- bind_cols(speaker_names, text_data) |>

```

```

## Creation de variables et nettoyage ##
mutate(year      = 2017,
       date       = ymd("2017-05-03"),
       country    = "France",
       id         = "Débat présidentiel 2017 second-tour",
       speaker_turn = 1:804,
       speaker = str_squish(str_replace_all(speaker, c("Mme" = "",
                                                         "^M." = "",
                                                         ":" = ""))),
       party = case_when(speaker == "Le Pen" ~ "Rassemblement national",
                         speaker == "Macron" ~ "Renaissance",
                         .default = "Journaliste")) |>
select(id, year, date, country, speaker, speaker_turn, party, text)

##Sauvegarde des données

write_csv(Data_debate_2017, "/Users/oliviasaffioti/Desktop/fas_1001_Saffioti/_tp/TP3/debat

# Faire notre recherche avec une analyse de dictionnaire

##Importer la base de données :

setwd("/Users/oliviasaffioti/Desktop/fas_1001_Saffioti/_tp/TP3")
getwd()

```

```
[1] "/Users/oliviasaffioti/Desktop/fas_1001_Saffioti/_tp/TP3"
```

```

debat_macron_lepen <- read.csv("debat_macron_lepen.csv")

## Libraries :

library(quanteda)

```

Package version: 3.3.1

Unicode version: 14.0

ICU version: 71.1

Parallel computing: 8 of 8 threads used.

See <https://quanteda.io> for tutorials and examples.

```
library(clessnverse)
```

DISCLAIMER: As of July 2023, `clessnverse` is no longer under active development.
To avoid breaking dependencies, the package remains available "as is" with no warranty of any kind.

```
library(tidyverse)
```

```
## Création du dictionnaire :
```

```
Dictionnaire_combiné <- list (Santé = c("santé", "médecin", "hôpital", "maladie", "patient"
```

```
## Nettoyer le texte : enlever la ponctuation et les majuscules du texte
```

```
debat_macron_lepen$text <- tolower(debat_macron_lepen$text)
```

```
debat_macron_lepen$text_clean <- gsub("[[:punct:]]", "", debat_macron_lepen$text)
```

```
## Créer une nouvelle base de données en sélectionnant les variables utiles
```

```
debat_macron_lepen <- debat_macron_lepen %>% select(speaker, text_clean, party) %>% filter(
```

```
## Appliquer le dictionnaire :
```

```
debat_macron_lepen_resultats <- run_dictionary(data = debat_macron_lepen, text = text_clean
```

100% expressions/words found

0.052 sec elapsed

```
## Faire un graphique en barres :
```

```
donnees_graphique <- debat_macron_lepen_resultats %>% group_by(speaker) %>% summarise(santé =
```

```
donnees_graphique_format_long <- tidyr::pivot_longer(donnees_graphique, cols = -speaker, names
```

```
Graphique <- ggplot(donnees_graphique_format_long, aes(x = catégorie, y = fréquence, fill =
```



```
print(Graphique)
```

