Categorizing Written Survey Responses with a Dictionary Approach

Katharine McCoy

2026-03-28

Contents

Introduction

This memo describes and provides code for a dictionary-based method for categorizing written free responses in the 2025 CES. It is intended to provide a detailed explanation of our methods and to enable another researcher to replicate our approach with similar data.

Background

The 2025 Campaign Period Survey was a survey of 20,180 respondents which asked about respondent's demographics as well as political and sociological attitudes, opinions, and behaviors. Respondents were asked in one question to write-in what the most important issue to them personally was for the 2025 federal election. Our goal was to put these free responses into broader categories, such as issues relating to the economy, environment, etc.

Approach

We approached the categorization task by first creating a list of keywords for each category we thought would be common. The keywords were generated by reviewing the literature political issue categories and identifying commonly used terms and phrases. For example, the environment dictionary started with words like "climate", "change", "oil", etc. These keywords formed the original dictionaries used.

We then ran the question responses through the dictionaries, and the number of words in a response that matched words in a specific dictionary was recorded. So for each category, a new variable was created which contains a number for each response corresponding to how many "hits" the response had for that category. We also created a dummy variable for each category which is entered as a 1 if there was any matches, and 0 if there were none. This means that a single response may be included in several categories.

Following the original categorization, we focused on responses that had not yet been categorized. These responses were pulled and sorted into alphabetical order so that common words could easily be seen. Common words were either added to existing category dictionaries if applicable, or a new category dictionary was

created. Additionally, common misspellings were added to the dictionaries. This was repeated a couple times until there were no more terms widely used across the un-categorized responses.

Note: the text to be searched was both in English and in French, so dictionaries contain both English and French words, as well as common misspellings of words in both languages. The final dictionaries used are included in the code below.

Acknowledgements

Many thanks to Thomas Galipeau, Nadjim Fréchet, and Maxime St-Jean for writing the previous drafts of the code.


```
## lading packages
# install.packages("tidyverse")
library(tidyverse)
# install.packages("haven")
library(haven)
# install.packages("magrittr")
library(magrittr)

# install.packages("tidytext")
library(tidytext)
# install.packages("quanteda")
library(quanteda)
# install.packages("tm")
library(tm)
# install.packages("SnowballC")
library(SnowballC)

# install.packages("crayon")
library(crayon)
# install.packages("progressr")
library(progressr)
# install.packages("progress")
library(progress)
# install.packages("lubridate")
library(lubridate)
# install.packages("quanteda.textstats")
library(quanteda.textstats)
# install.packages("tictoc")
library(tictoc)

# install.packages("gtsummary")
library(gtsummary)
```

```
## Functions
runDictionary <- function(
    dataA, # input database where the search will be performed
    word, # name of the column to be searched
    dictionaryA) { # dictionary containing terms to be searched for
  tictoc::tic() # starts a timer (speedtesting the function)
  dataA <- dataA %>%
    mutate(word = {{word}}) # creates a copy of the input column called "word"
```

```
  corpusA <- tokens(dataA$word) # tokenizes text in the "word" column
  dfmA <- dfm(tokens_lookup(corpusA, # checks for frequency of each token
                            dictionaryA,
                            nested_scope = "dictionary"))

## Progress bar ##
pb <- progress_bar$new(
  format = yellow(" downloading [:bar] :percent in :elapsed"),
  total = 100, clear = FALSE, width= 60)
purrr::walk(1:100, ~{pb$tick(); Sys.sleep(0.01)})
message(green("100% expressions/words found"))
tictoc::toc() # ends timer

dataB <- convert(dfmA, # puts the frequency count as a data.frame
                 to = "data.frame")
return(dataB) # returns the count
}
```

```
# Makes the text lowercase, removes some special characters and punctuation
ces25$cps25_imp_iss <- stringr::str_replace_all(ces25$cps25_imp_iss, "[A-Z]",
                                                ~tolower(.x))
ces25$cps25_imp_iss <- sub("^(\\w+)\\s+(\\w+)$", "\\2 \\1", ces25$cps25_imp_iss)
ces25$cps25_imp_iss <- gsub("[[:punct:]]", "", ces25$cps25_imp_iss)
```

For each of the categories, we do the same workflow. This is a commented version of the first category, the economy, for your reference.

```
## Workflow - Economy
dictionaryeco <- dictionary( # creates the category-specific dictionary
  list(economy = c("economy", "jobs", "employment", "tax", "taxs", "taxes",
                   "job", "conomie", "con", "l'conomie", "economie",
                   "conomique", "dette", "debt", 'deficit', "impts", "finances",
                   "finance", "impot", "dficits", "budget", "conomiques",
                   "economics", "balanced", "dollars", "deficits", "evonomy",
                   "low income", "spending", "trade", "depenses", "déficit",
                   "dpense", "middle class", "taxing", "wage", "wages",
                   "economic", "conomiques", "budgets", "taxation",
                   "fiscal", "market", "recession", "growth", "loans",
                   "dollars", "budgétaire", "leconomie", "argent",
                   "l'endettement", "living", "cost", "money", "spending",
                   "inequality", "prices", "trade", "inflation", "poor",
                   "enconomie", "ecomomie", "econamy", "emploi", "ecomomy",
                   "econics", "unemployment", "impots", "affordability",
                   "d'impot", "d'impo", "d'impt", "emploie", "economique",
                   "ecomony", "work", "unemployed", "taxe", "taxed", "dficit",
                   "financial", "budgtaire","l'economie", "economist",
                   "économiste", "cost of", "living", "coût", "prix", "cost",
                   "léconomie", "cout", "expensive", "industry", "industrie",
                   "commerciale", "Économie", "afforability", "affordability",
                   "afford", "financière", "financial", "largent",
                   "rising cost", "léconomique", "impôts", "dimpôts",
                   "investments", "investment", "investissements", "économique",
                   "income", "lecout", "trading")))
```

```r
# this finds those who mentioned one of the economic words in the 2025 CES

ces25.econ <- runDictionary( # this creates a dataframe with a column counting
  dataA = ces25, # the number of times a word in the dictionary
  word = cps25_imp_iss, # is mentioned.
  dictionaryA = dictionaryeco)

ces25$economy <- ces25.econ$economy # puts the column into the original data

ces25 <- ces25 %>%
  mutate(economy.dum = ifelse(economy >= 1, 1, 0)) # adds a binary column
```

```r
## Workflow - Environment

dictionaryenviro <-dictionary(
  list(enviro = c("climate", "change", "envi", "pipelines", "oil", "carbon",
                  "pipeline", "environnement", "environmental", "environment",
                  "climate change", "l'environnement", "warming",
                  "l'environnement", "climatiques", "lenvironement", "ges",
                  "rechauffement", "gas", "enviroment", "water",
                  "sustainability", "enviromental", "environnement",
                  "enviroment", "écologie", "l'envéronnement", "l'ecologie",
                  "l'environnemen", "l'environnemenr", "l'environnent",
                  "emvironnement", "environnemen5", "ecology", "co2", "polluer",
                  "pollute", "pollution", "planet", "nergtiques", "energy",
                  "carbone", "greener", "green", "climatique", "environnment",
                  "enviournment", "climat", "envioroment", "earth", "cologie",
                  "environnementaux", "ecologie", "enviornment", "enviro",
                  "enviormental", "enironment", "fossiles", "fossil",
                  "environement", "environmentalism", "l'cologie",
                  "l'environement", "pipe", "lenvironnement", "lenrironnement",
                  "environnemental")))

ces25.enviro <- runDictionary(dataA = ces25,
                              word = cps25_imp_iss,
                              dictionaryA = dictionaryenviro)

ces25$enviro <- ces25.enviro$enviro

ces25 <- ces25 %>%
  mutate(enviro.dum = ifelse(enviro >= 1,1,0))
```

```r
## Workflow - Immigration

dictionaryimmigration <- dictionary(
  list(immigration = c("immigration", "illgale", "illégale", "minority",
                       "discrimination", "immigrants", "immigrant", "langue",
                       "l'imigration", "d'immigrant", "foreign", "immigrations",
                       "imagration", "imigration", "immegrants",
                       "l'immigration", "emigration", "refugee", "refugees",
                       "immagration", "immgration", "imigrant", "limmigration")))

ces25.immigration <- runDictionary(dataA = ces25,
```

```
                                    word = cps25_imp_iss,
                                    dictionaryA = dictionaryimmigration)

ces25$immigration <- ces25.immigration$immigration

ces25 <- ces25 %>%
  mutate(immigration.dum = ifelse(immigration >= 1, 1, 0))
```

```
## Workflow - Healthcare

dictionnaryhealthcare <- dictionary(
  list(healthcare = c("health", "health-care", "care", "sant", "soins", "life",
                      "mental", "disability","pharmacare", "disabled", "drugs",
                      "drug", "medicare", "santé", "medical", "heath",
                      "prescriptions", "doctors", "sante", "santé", "soin",
                      "santè", "docteur", "medical", "healthcare", "healtcare",
                      "heathcare", "hospitals", "medicine", "bien être",
                      "wellbeing", "hralthcare", "medicade", "medicaid")))

ces25.healtchcare <- runDictionary(dataA = ces25,
                                    word = cps25_imp_iss,
                                    dictionaryA = dictionnaryhealthcare)

ces25$healthcare <- ces25.healtchcare$healthcare

ces25 <- ces25 %>%
  mutate(healthcare.dum = ifelse(healthcare >= 1, 1, 0))
```

```
## Workflow - Housing

dictionnaryhousing <- dictionary(
  list(housing = c("housing", "affordable", "rent", "homeless", "rental",
                   "unaffordable", "renting","home", "homes", "dwelling",
                   "loyer", "maisons", "sans-abris", "logement", "logements",
                   "rents", "homelessness", "housingaffordability",
                   "itinérance", "inflationhousingrenting")))

ces25.housing <- runDictionary(dataA = ces25,
                                word = cps25_imp_iss,
                                dictionaryA = dictionnaryhousing)

ces25$housing <- ces25.housing$housing

ces25 <- ces25 %>%
  mutate(housing.dum = ifelse(housing >= 1, 1, 0))
```

```
## Workflow - Seniors

dictionnaryseniors <- dictionary(
  list(seniors = c("pension", "pensions", "seniors", "senior", "aines", "ages",
                   "cpp", "elderly", "oas", "aging", "senior's", "retirement",
                   "âgées", "ainés", "65", "vieillisse", "viellesse",
                   "vielliesse", "vieux", "ainees", "aine", "vieillissement",
```

```
                  "veillesse", "pesion", "age", "old people", "retirees",
                  "retraite", "retired", "senoir")))

ces25.seniors <- runDictionary(dataA = ces25,
                               word = cps25_imp_iss,
                               dictionaryA = dictionnaryseniors)

ces25$seniors <- ces25.seniors$seniors

ces25 <- ces25 %>%
  mutate(seniors.dum = ifelse(seniors >= 1, 1,0))
```

```
## Workflow - Leaders

dictionnaryleaders <- dictionary(
  list(leaders = c("carney", "mark", "libéral", "libral", "liberals",
                "leadership", "leader", "justin", "conservatives", "parties",
                "leaders", "pm", "andrew", "sheer", "singh", "blanchet", "ndp",
                "bloc", "green", "paul", "may", "otoole", "trudeau", "justin",
                "toole", "bernier", "pm", "politician", "trudeau's", "o'toole",
                "libéraux", "leader", "ford", "scheer", "prime minister",
                "candidate", "candidates", "liberal", "pierre", "poilievre",
                "conservateurs", "conservateur", "carnay", "marc",
                "conservatrices", "poliviere", "conservative")))

ces25.leaders <- runDictionary(dataA = ces25,
                               word = cps25_imp_iss,
                               dictionaryA = dictionnaryleaders)

ces25$leaders <- ces25.leaders$leaders

ces25 <- ces25 %>%
  mutate(leaders.dum = ifelse(leaders >= 1, 1, 0))
```

```
## Workflow - Ethics

dictionnaryethics <- dictionary(
  list(ethics = c("gouvernement", "corruption", "honesty", "ethics",
                  "transparency", "accountability", "responsibility", "truth",
                  "lies", "lying", "ethical", "transparent", "integretity",
                  "corrupt", "trustworthy", "dishonesty", "liar",
                  "transparence", "moral", "integrity", "honest", "trust",
                  "corruptions", "coruption", "credibility", "greed",
                  "promesses", "honestly", "honnetete", "honntet", "morality",
                  "moral", "morals", "accountable", "accountibility",
                  "rights", "represented", "fairness", "governance",
                  "responsible government")))

ces25.ethics <- runDictionary(dataA = ces25,
                              word = cps25_imp_iss,
                              dictionnaryethics)

ces25$ethics <- ces25.ethics$ethics
```

```r
ces25 <- ces25 %>%
  mutate(ethics.dum = ifelse(ethics >=1,1,0))
```

```r
## Workflow - Education

dictionnaryeducation <- dictionary(
  list(education = c("education", "ducation", "school", "schools",
                     "educational", "university", "tuition", "student",
                     "students", "schooling", "l'ducation", "l'education",
                     "deducation", "Éducation")))

ces25.education <- runDictionary(dataA = ces25,
                                 word = cps25_imp_iss,
                                 dictionnaryeducation)

ces25$education <- ces25.education$education

ces25 <- ces25 %>%
  mutate(education.dum = ifelse(education >=1,1,0))
```

```r
## Workflow - Crime and Guns

dictionnarycrime <- dictionary(
  list(crime = c("crime", "crimes", "criminal", "criminals", "gang", "gangs",
                 "safety", "gun", "firearm", "violence", "illegal", "cop")))

ces25.crime <- runDictionary(dataA = ces25,
                             word = cps25_imp_iss,
                             dictionnarycrime)

ces25$crime <- ces25.crime$crime

ces25 <- ces25 %>%
  mutate(crime.dum = ifelse(crime >=1,1,0))
```

```r
## Workflow - Indigenous

dictionnaryindigenous <- dictionary(
  list(indigenous = c("indigenous", "aboriginal", "reconciliation",
                      "first nations", "first nation", "indeginous", "native")))

ces25.indigenous <- runDictionary(dataA = ces25,
                                  word = cps25_imp_iss,
                                  dictionnaryindigenous)

ces25$indigenous <- ces25.indigenous$indigenous

ces25 <- ces25 %>%
  mutate(indigenous.dum = ifelse(indigenous >=1,1,0))
```

```r
## Workflow - Other Welfare
```

```
dictionnarywelfare <- dictionary(
  list(welfare = c("childcare", "children", "daycare", "dental", "welfare",
                   "social services", "social programs", "social service",
                   "social program", "family", "families", "famille", "child",
                   "children's", "basic income", "familiale", "familles",
                   "poverty", "social assistance", "public", "pauvret",
                   "service", "services", "parental", "redistribution",
                   "aide sociale", "social aid")))

ces25.welfare <- runDictionary(dataA = ces25,
                               word = cps25_imp_iss,
                               dictionnarywelfare)

ces25$welfare <- ces25.welfare$welfare

ces25 <- ces25 %>%
  mutate(welfare.dum = ifelse(welfare >=1,1,0))
```

```
## Workflow - Electoral Reform

dictionnaryelection <- dictionary(
  list(election = c("election", "electoral", "voting", "voter",
                    "representation", "democracy", "first past the post",
                    "proportional", "vote")))

ces25.election <- runDictionary(dataA = ces25,
                                word = cps25_imp_iss,
                                dictionnaryelection)

ces25$election <- ces25.election$election

ces25 <- ces25 %>%
  mutate(election.dum = ifelse(election >=1,1,0))
```

```
## Workflow - Women's Issues and Abortion

dictionnarywomen <- dictionary(
  list(women = c("women", "women's", "abortion", "abortions", "woman",
                 "woman's", "unborn", "reproductive", "femme", "femmes",
                 "gender", "maternity", "womens", "anatiabortion")))

ces25.women <- runDictionary(dataA = ces25,
                             word = cps25_imp_iss,
                             dictionnarywomen)

ces25$women <- ces25.women$women

ces25 <- ces25 %>%
  mutate(women.dum = ifelse(women >=1,1,0))
```

```
## Workflow - Security / Defense and Intarional Relations

dictionnarysecurity <- dictionary(
```

```
  list(security = c("security", "defense", "international", "china", "defence",
                    "war", "wars", "relations", "global", "israel", "u.s.",
                    "segurity", "scurit", "military", "palestin", "palestine",
                    "defense", "interference", "ingérence", "défense",
                    "terrorism", "sécurité", "géopolitiques", "armes",
                    "weapons", "gaza")))

ces25.security <- runDictionary(dataA = ces25,
                                word = cps25_imp_iss,
                                dictionnarysecurity)

ces25$security <- ces25.security$security

ces25 <- ces25 %>%
  mutate(security.dum = ifelse(security >=1,1,0))
```

```
## Workflow - Quebec

dictionnaryquebec <- dictionary(
  list(quebec = c("quebec", "21", "qubec", "francophone", "lacit","laicit",
                  "laicite", "québec", "québécois")))

ces25.quebec <- runDictionary(dataA = ces25,
                              word = cps25_imp_iss,
                              dictionnaryquebec)

ces25$quebec <- ces25.quebec$quebec

ces25 <- ces25 %>%
  mutate(quebec.dum = ifelse(quebec >=1,1,0))
```

```
## Workflow - Race

dictionnaryrace <- dictionary(
  list(race = c("race", "racism", "black", "white", "antisemitism",
                "islamophoby", "islamaphobia")))

ces25.race <- runDictionary(dataA = ces25,
                            word = cps25_imp_iss,
                            dictionnaryrace)

ces25$race <- ces25.race$race

ces25 <- ces25 %>%
  mutate(race.dum = ifelse(race >=1,1,0))
```

```
## Workflow - Trump
dictionnarytrump <- dictionary(
  list(trump = c("trump", "turmp", "donald", "president", "président",
                 "trumps", "dtrump", "bufoontrump", "trumps", "usatrump")))

ces25.trump <- runDictionary(dataA = ces25,
                             word = cps25_imp_iss,
```

```
                              dictionnarytrump)

ces25$trump <- ces25.trump$trump

ces25 <- ces25 %>%
  mutate(trump.dum = ifelse(trump >=1,1,0))

## Workflow - Tariff

dictionnarytariff <- dictionary(
  list(tariff = c("tariff", "tariffs", "tarif", "tarifs", "economist",
                  "économiste", "big beautiful bill", "tarriffs", "terrifs",
                 "economytariffs", "tarrifs", "tarriffs", "taxestariffs",
                 "thariff")))

ces25.tariff <- runDictionary(dataA = ces25,
                              word = cps25_imp_iss,
                              dictionnarytariff)

ces25$tariff <- ces25.tariff$tariff

ces25 <- ces25 %>%
  mutate(tariff.dum = ifelse(tariff >=1,1,0))

## Workflow - Borders

dictionnaryborders <- dictionary(
  list(borders = c("soveriegnty", "border", "souveraineté", "frontière",
                   "annex", "annexation", "annexion", "trump", "sovereignty",
                   "us", "usa", "states", "the states", "america", "americans",
                   "états-unis", "américain", "americain", "américains",
                   "americains", "les état unis", "frontières", "indépendance",
                   "independence", "sovreignty", "independent", "indépendant",
                   "États unis", "lintégrité territoriale", "autonomie",
                   "autonomy", "51st", "canadian identity", "américaines",
                   "etats unis", "keeping canada", "american", "souveraineté",
                   "neighbours")))

ces25.borders <- runDictionary(dataA = ces25,
                              word = cps25_imp_iss,
                              dictionnaryborders)

ces25$borders <- ces25.borders$borders

ces25 <- ces25 %>%
  mutate(borders.dum = ifelse(borders >=1,1,0))

## Workflow - Trump, Tariffs and Borders combined

dictionnarycombined <- dictionary(
  list(combined = c("trump", "turmp", "donald", "president", "président",
                    "tariff", "tariffs", "tarif", "tarifs", "economist",
                   "économiste", "big beautiful bill", "soveriegnty",
```

```
                    "border", "souveraineté", "frontière", "annex", "annexation",
                    "annexion", "trump", "sovereignty", "us", "usa", "states",
                    "the states", "america", "americans", "états-unis",
                    "américain", "americain", "américains", "americains",
                    "les état unis", "Étasunis", "Étatsunis", "frontières",
                    "indépendance","independence", "sovreignty", "terrifs",
                    "independent", "indépendant", "États unis", "trumps",
                    "lintégrité territoriale", "economytariffs", "autonomie",
                    "autonomy", "51st", "tarrifs", "canadian identity",
                    "américaines", "etats unis", "tarriffs", "taxestariffs",
                    "thariff", "keeping canada", "american", "usatrump",
                    "souverainté", "neighbours")))

ces25.combined <- runDictionary(dataA = ces25,
                                word = cps25_imp_iss,
                                dictionnarycombined)

ces25$combined <- ces25.combined$combined

ces25 <- ces25 %>%
  mutate(combined.dum = ifelse(combined >=1,1,0))
```

```
## Workflow - Nonanswers

dictionnaryidk <- dictionary(
  list(idk = c("99", "unsure", "don't know", "je ne sais pas", "idk",
               "prefer not", "nothing", "no", "don't know", "jw sais pas",
               "je sais pas", "je ne saos pas", "je ne connais", "je bsais pas",
               "jai pas de reponse")))

ces25.idk <- runDictionary(dataA = ces25,
                                word = cps25_imp_iss,
                                dictionaryA = dictionnaryidk)

ces25$idk <- ces25.idk$idk

ces25 <- ces25 %>%
  mutate(idk.dum = ifelse(idk >= 1, 1, 0))
```

```
## Summary of the results

ces25 %>%
  tbl_summary(include = c(economy.dum, enviro.dum, immigration.dum,
                          healthcare.dum, housing.dum, seniors.dum, leaders.dum,
                          ethics.dum, education.dum, crime.dum, indigenous.dum,
                          welfare.dum, election.dum, women.dum, security.dum,
                          idk.dum, quebec.dum, race.dum, combined.dum,
                          tariff.dum,trump.dum, borders.dum),
              label = list(economy.dum ~ "The Economy",
                           enviro.dum ~ "The Environment",
                           immigration.dum = "Immigration",
                           healthcare.dum ~ "Healthcare",
                           housing.dum ~ "Housing", seniors.dum ~ "Seniors",
```

Characteristic	**N = 20,180**[1]
The Economy	7,085 (35%)
The Environment	653 (3.2%)
Immigration	686 (3.4%)
Healthcare	1,258 (6.2%)
Housing	1,194 (5.9%)
Seniors	188 (0.9%)
Party Leaders	802 (4.0%)
Ethical Concerns	770 (3.8%)
Education	125 (0.6%)
Crime	275 (1.4%)
Indigenous issues and Reconciliation	29 (0.1%)
Welfare	248 (1.2%)
Electoral Reform	285 (1.4%)
Women's Issues	154 (0.8%)
Security and International Relations	640 (3.2%)
Don't know / did not answer	2,974 (15%)
Quebec	90 (0.4%)
Race	45 (0.2%)
US Relations, Trump and Tariffs	4,016 (20%)
Tariffs	1,184 (5.9%)
Trump	1,629 (8.1%)
US Relations	3,215 (16%)

[1] n (%)

```
                    leaders.dum ~ "Party Leaders",
                    ethics.dum ~ "Ethical Concerns",
                    education.dum ~ "Education", crime.dum ~ "Crime",
                    indigenous.dum ~ "Indigenous issues \n and Reconciliation",
                    welfare.dum ~ "Welfare",
                    election.dum ~ "Electoral Reform",
                    women.dum ~ "Women's Issues", security.dum ~
                      "Security and \n International Relations",
                    idk.dum ~ "Don't know / did not answer",
                    quebec.dum ~ "Quebec", race.dum ~ "Race",
                    combined.dum ~ "US Relations, Trump and Tariffs",
                    tariff.dum ~ "Tariffs", trump.dum ~ "Trump",
                    borders.dum ~ "US Relations"))
```