# Koronavírus

Granát Marcell és Mazzag Bálint

2021. március 1.

## Tartalomjegyzék

**Absztrakt**

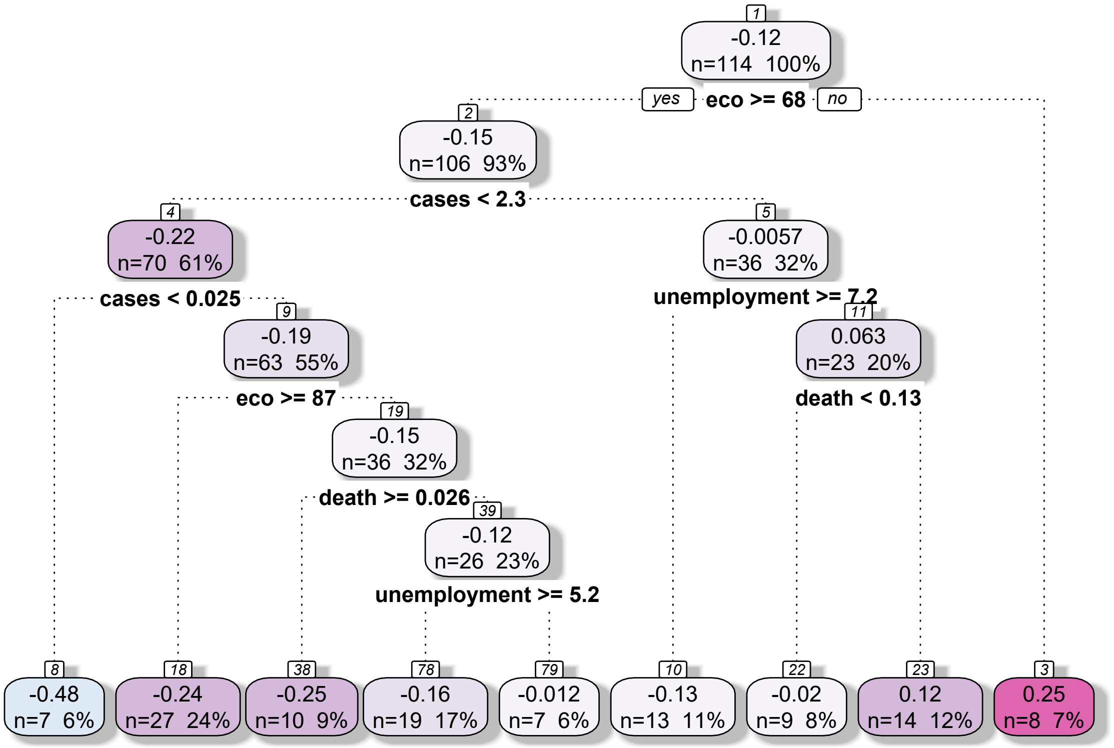
Here is the abstract.

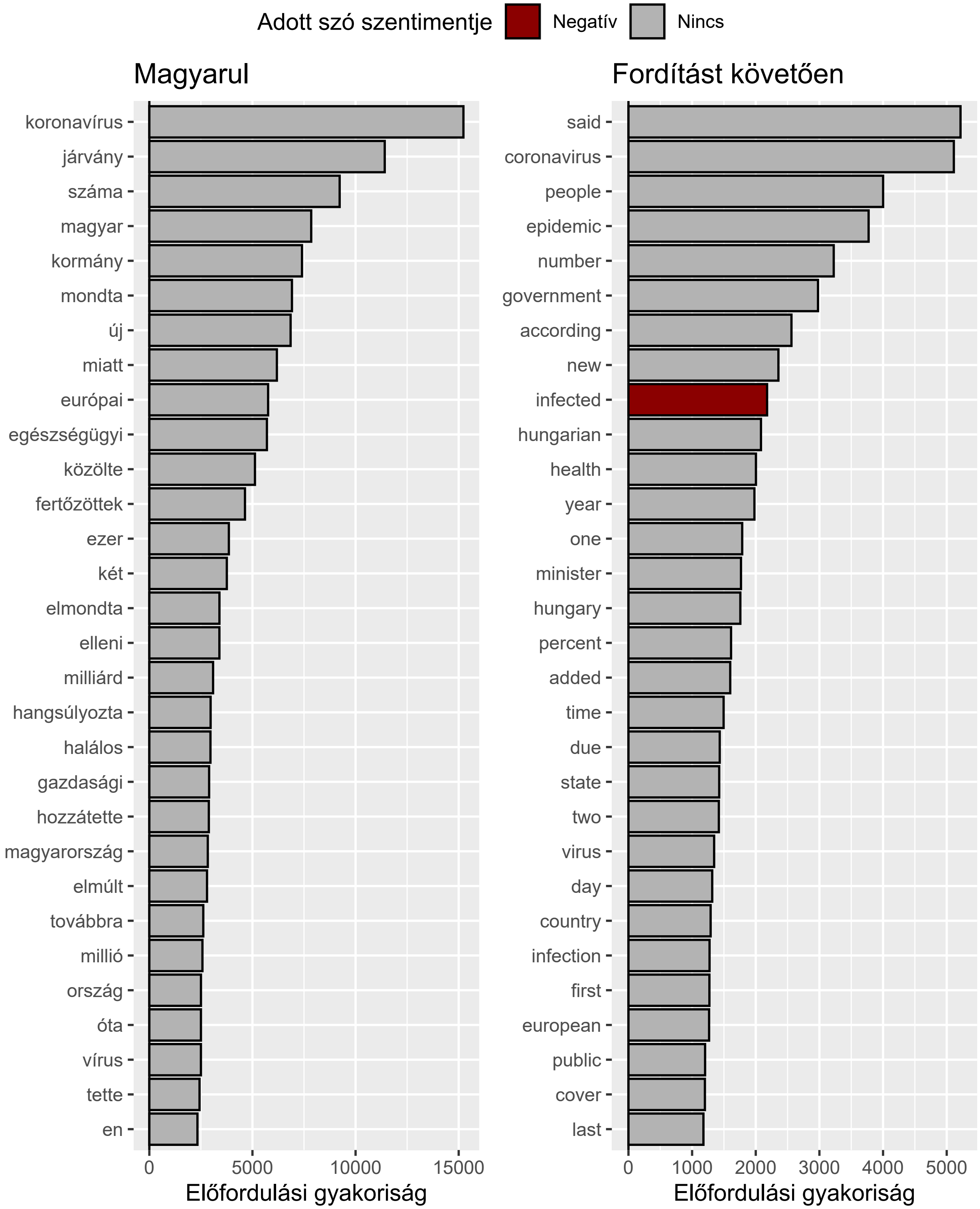
# Bevezetés

# Adatok

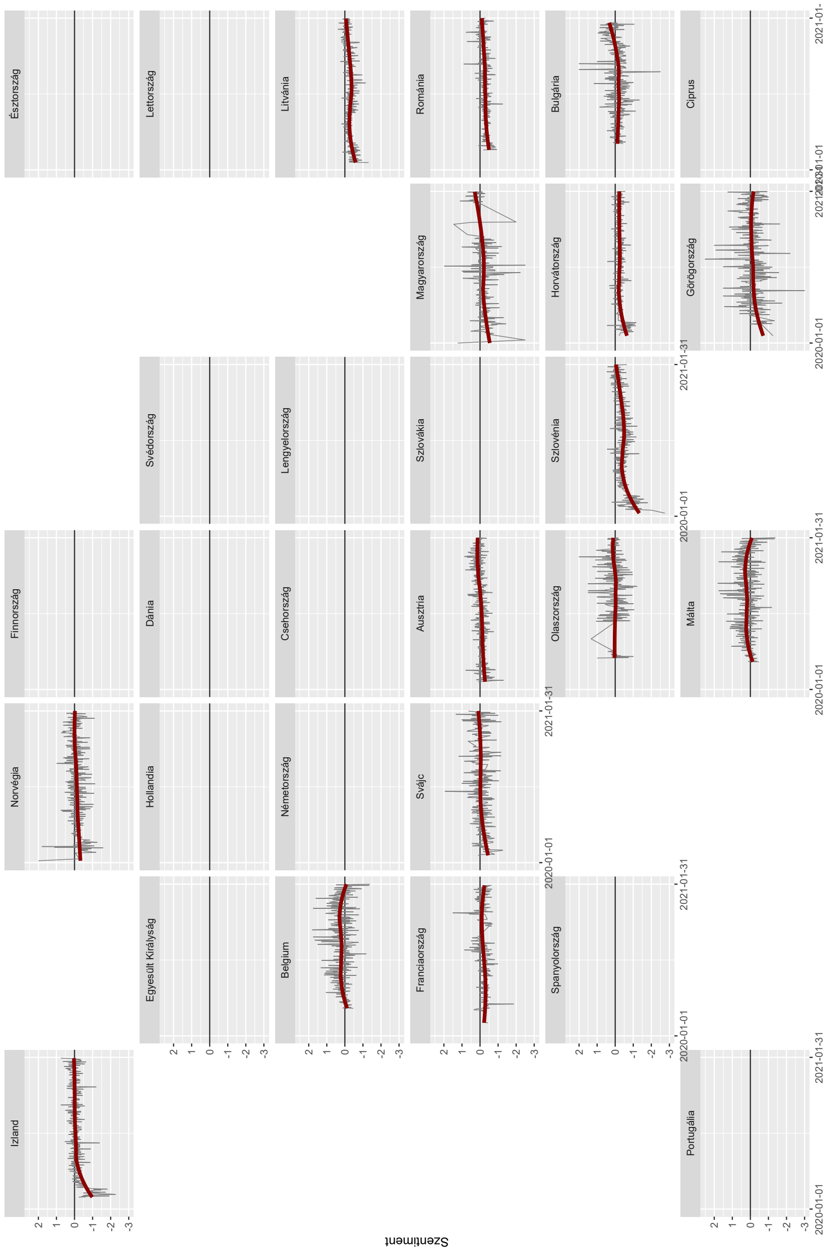
## Gépi fordítás

# Leíró statisztikák

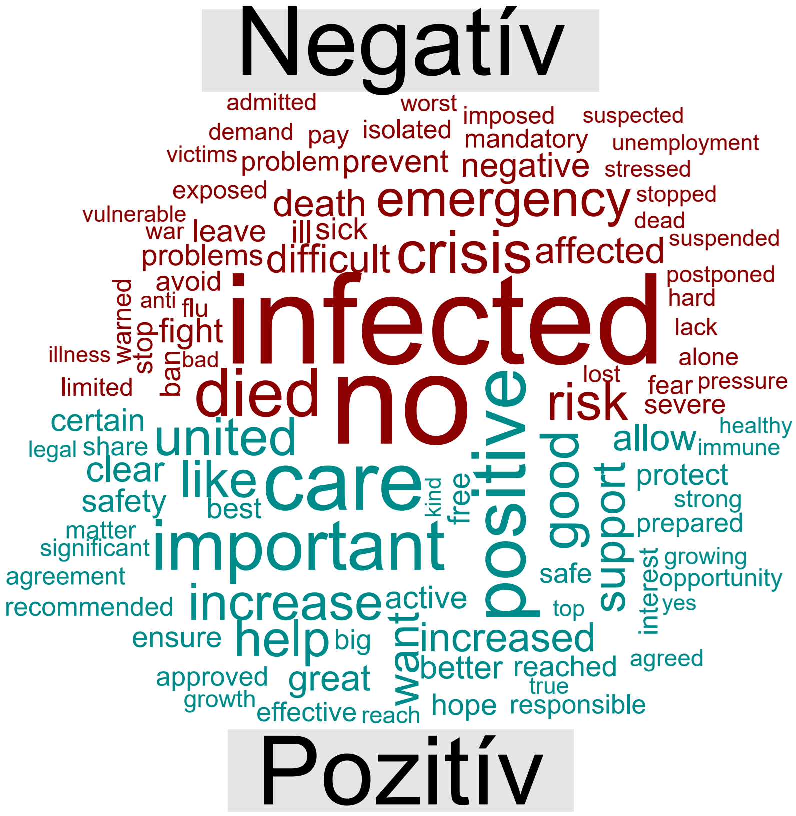




1. ábra. Leggyakrabban előforduló szavak a magyar nylevű cikkekben a fordítást megelőzően és azt követően.



2. ábra. A szentiment alakulása országonként



3. ábra. Leggyakrabban előforduló pozitív és negatív szentimenttel rendelkező szavak

# Hivatkozások

Adsera, A. (2004), Changing fertility rates in developed countries. the impact of labor market institutions', *Journal of population economics* **17**(1), 17–43.

# Függelék: R kódok

```r
# Set up --------------------------------------------------------------------------------

## Packages =============================================================================

library(tidyverse)
library(patchwork)
library(knitr)
library(broom)
library(geofacet)
library(tidytext)
library(tm)
library(wordcloud)

## Gg theme =============================================================================

update_geom_defaults("point", list(fill = "cyan4",
                                   shape = 21,
                                   color = "black",
                                   size = 1.4))
update_geom_defaults("line",
                     list(color = "midnightblue", size = 1.4))

update_geom_defaults("smooth", list(color = "red4", size = 1.4))

update_geom_defaults("density",
                     list(color = "midnightblue", fill =  "midnightblue",
                          alpha = .3, size = 1.4))

extrafont::loadfonts(device="win")

theme_set(theme_grey() + theme(
  legend.direction = "vertical",
  plot.caption = element_text(family = "serif")
))

# Data ----------------------------------------------------------------------------------

# Articles ==============================================================================

load("dat.RData")
# This RData contains the articles after the main cleaning process
# To ensure full reproducibility see the attached files at the corresponding
# GitHub Repo: -> https://github.com/MarcellGranat/CoronaSentiment <-

Hungary_rawtext <- readxl::read_excel("scrapping raw csv/Hungary_rawtext.xlsx") %>%
  # Hungarian articles before translation
  select(date, title, URL = links, text) %>%
  mutate_all(function(x) str_remove_all(x, "\r")) %>%
  mutate_all(function(x) str_remove_all(x, "\t")) %>%
  mutate_all(function(x) str_remove_all(x, "\n")) %>%
  mutate_at(-1, function(x) zoo::na.locf(x)) %>%
  filter(!str_detect(date, '_x000') & date != '0') %>%
  filter(!str_detect(text, 'mtva_player')) %>% # TODO consider a better solution
  mutate(
    date = gsub(" -.*", "", date),
    text = str_remove_all(text, "_x000D_"),
    date = lubridate::ymd(date)
  ) %>%
  tidytext::unnest_tokens(words, text)

### Add sentiment values to our data ####################################################

dat_sentiment <- dat %>%
  select(date, text, country) %>%
  mutate(country = ifelse(str_detect(country, "BE"), "BE", country)) %>%
  {left_join(tidytext::unnest_tokens(., words, text),
             get_sentiments("afinn"), by=c("words"="word"))}
# TODO other packages

dat_sentiment_daily <- dat_sentiment %>%
  group_by(date, country) %>%
  summarise(value = mean(value, na.rm = T), n = n()) %>%
  ungroup() %>%
  na.omit() %>%
  rename(code = country)

dat_sentiment_monthly <- dat_sentiment %>%
  na.omit() %>%
  mutate(
    date = lubridate::ym(paste(lubridate::year(date), lubridate::month(date), sep = "-"))
  ) %>%
  group_by(date, country) %>%
  summarise(value = mean(value, na.rm = T), n = n()) %>%
  ungroup() %>%
  na.omit() %>%
  rename(code = country)

# COVID data ============================================================================

dat_covid <-
  readr::read_csv("https://covid.ourworldindata.org/data/owid-covid-data.csv") %>%
  transmute(code = countrycode::countrycode(iso_code, origin = 'iso3c',
                                            destination = 'iso2c'),
            date,
            cases = new_cases_per_million/1000,
            death = new_deaths_per_million/1000
  )

dat_covid_monthly <- dat_covid %>%
  mutate(
    date = lubridate::ym(paste0(lubridate::year(date), '-', lubridate::month(date)))
  ) %>%
  group_by(date, code) %>%
  summarise(cases = sum(cases, na.rm = T), death = sum(death, na.rm = T)) %>%
  ungroup()
```

```r

# Data from Eurostat ====================================================================

dat_eco_sent <- eurostat::get_eurostat('ei_bssi_m_r2')
# Economic sentiment indicator

dat_unemployment <- eurostat::get_eurostat("une_rt_m") %>%
# unemployment
  filter(age == "TOTAL", sex == "T", s_adj == "NSA", unit == "PC_ACT") %>%
  select(date = time, code = geo, unemployment = values)
# Grid to facet_geo =====================================================================

mygrid <- data.frame(
  row = c(5, 1, 1, 1, 1, 2, 2, 2, 2, 2, 3, 3, 3, 3, 3, 4, 4, 4, 4, 4, 4, 5, 5, 5, 5, 6,
          6, 6, 6),
  col = c(7, 1, 3, 4, 7, 7, 5, 4, 2, 3, 7, 2, 3, 5, 4, 4, 7, 6, 2, 5, 3, 6, 4, 5, 2, 4
          , 7, 1, 6),
  code = c("BG", "IS", "NO", "FI", "EE", "LV", "SE", "DK", "UK", "NL", "LT", "BE", "DE",
           "PL", "CZ", "AT", "RO", "HU", "FR", "SK", "CH", "HR", "IT", "SI", "ES", "MT",
           "CY", "PT", "EL"),
  name = c("Bulgária", "Izland", "Norvégia", "Finnország", "Észtország", "Lettország",
           "Svédország", "Dánia", "Egyesült Királyság", "Hollandia", "Litvánia",
           "Belgium", "Németország", "Lengyelország", "Csehország", "Ausztria",
           "Románia", "Magyarország", "Franciaország", "Szlovákia", "Svájc",
           "Horvátország", "Olaszország", "Szlovénia", "Spanyolország", "Málta", "Ciprus",
           "Portugália", "Görögország"),
  stringsAsFactors = FALSE
)

# Automatic translation =================================================================

st_hu <- c(stopwords::stopwords('hungarian'), "is", "ha", "hozzá", "címlapfotó",
           "illusztráció") %>%
  {ifelse(str_starts(., "új"), NA, .)} %>%
  na.omit()

ggpubr::ggarrange(
  Hungary_rawtext %>%
    filter(!str_detect(words, '\\d')) %>%
    anti_join(data.frame(words = st_hu)) %>%
    count(words, sort = T) %>%
    arrange(desc(n)) %>%
    head(30) %>%
    mutate(
      words = fct_reorder(words, n)
    ) %>%
    ggplot() +
    aes(n, words) +
    geom_vline(xintercept = 0) +
    geom_col(color = 'black', fill = "gray70") +
    labs(title = 'Magyarul', x = 'Előfordulási gyakoriság', y = NULL),

  dat_sentiment %>%
    filter(country == 'HU') %>%
    filter(!str_detect(words, '\\d')) %>%
    anti_join(data.frame(words = c(stopwords::stopwords(), "also", "can"))) %>%
    count(words, value, sort = T) %>%
    arrange(desc(n)) %>%
    head(30) %>%
    mutate(
      value = case_when(
        value < 0 ~ "Negatív",
        value > 0 ~ "Pozitív",
        T ~ "Nincs"
      ),
      words = fct_reorder(words, n)
    ) %>%
    ggplot() +
    aes(n, words, fill = value) +
    geom_vline(xintercept = 0) +
    geom_col(color = "black") +
    labs(title = 'Fordítást követően', x = 'Előfordulási gyakoriság', y = NULL,
         fill = "Adott szó szentimentje") +
    scale_fill_manual(values = c('red4', 'gray70', 'green')) +
    theme(
      legend.position = 'bottom',
      legend.direction = 'horizontal'
    ), common.legend = T
)

# Explore the data ----------------------------------------------------------------------

ggplot(dat_sentiment_daily, aes(date, value)) +
  geom_hline(yintercept = 0, color = "grey20") +
  geom_line(size = .3, color = 'grey50') +
  geom_smooth(size = 1.5, se = F) +
  facet_geo(~ code, grid = mygrid, label = 'name') +
  scale_x_date(limits = c(min(dat_sentiment_daily$date), max(dat_sentiment_daily$date)),
               breaks = c(min(dat_sentiment_daily$date), max(dat_sentiment_daily$date))) +
  labs(y = "Szentiment", x = NULL)

library(reshape2)

dat_sentiment %>%
  na.omit() %>%
  mutate(
    sentiment = ifelse(value > 0, "Pozitív", "Negatív")
  ) %>%
  count(words, sentiment, sort = TRUE) %>%
  acast(words ~ sentiment, value.var = "n", fill = 0) %>%
  comparison.cloud(colors = c("red4", "cyan4"),
                   max.words = 100)

dat_plm <- dat_eco_sent %>%
  filter(indic == "BS-ESI-I") %>%
  select(date = time, code = geo, eco = values) %>%
  merge(dat_sentiment_monthly) %>%
  merge(dat_unemployment) %>%
  merge(dat_covid_monthly) %>%
  mutate(
    t = lubridate::interval(lubridate::ymd('2020-01-01'), date),
    t = lubridate::as.period(t) %/% months(1)
  )

# Regression tree -----------------------------------------------------------------------

m_tree <- rpart::rpart(data = dat_plm, formula = value ~ .-date-code-n,
                       cp = .01)

rattle::fancyRpartPlot(m_tree, palettes = 'PuRd', sub = NULL)
```