

# SkryptR.R

mp458994

2025-05-30

```
#install.packages(c("tm", "tidytext", "tidyverse", "ggplot2", "textdata", "ggthemes", "SentimentAnalysis",  
"tidyr", "tokenizers", "dplyr"))
```

```
options(repos = c(CRAN = "https://cran.rstudio.com"))
```

```
library("tidyverse")
```

```
## Warning: pakiet 'tidyverse' został zbudowany w wersji R 4.4.3
```

```
## Warning: pakiet 'tidyr' został zbudowany w wersji R 4.4.3
```

```
## Warning: pakiet 'purrr' został zbudowany w wersji R 4.4.3
```

```
## Warning: pakiet 'dplyr' został zbudowany w wersji R 4.4.3
```

```
## Warning: pakiet 'stringr' został zbudowany w wersji R 4.4.3
```

```
## Warning: pakiet 'forcats' został zbudowany w wersji R 4.4.3
```

```
## Warning: pakiet 'lubridate' został zbudowany w wersji R 4.4.3
```

```
## -- 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 3.5.1 v tibble 3.2.1
```

```
## v lubridate 1.9.4 v tidyr 1.3.1
```

```
## v purrr 1.0.4
```

```
## -- 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 errors
```

```
library("ggplot2")
```

```

library("ggthemes")

## Warning: pakiet 'ggthemes' został zbudowany w wersji R 4.4.3

library(SentimentAnalysis)

## Warning: pakiet 'SentimentAnalysis' został zbudowany w wersji R 4.4.3

##
## Dołączanie pakietu: 'SentimentAnalysis'
##
## Następujący obiekt został zakryty z 'package:base':
##
## write

library("tokenizers")

## Warning: pakiet 'tokenizers' został zbudowany w wersji R 4.4.3

library("tidyr")
library("dplyr")

#wybór folderu z transkryptami wypowiedzi
folder <- "Wypowiedzi"

#wczytanie listy plików
file_list <- list.files(path = folder, pattern = "\\\\.txt$", full.names = TRUE);
n <- length(file_list);

#wczytanie samych plików do odpowiedniej ramki danych
teksty <- data.frame(stringsAsFactors = FALSE);

for (i in 1:n) {
  name <- tools::file_path_sans_ext(basename(file_list[i]));
  text <- paste(readLines(file_list[i], encoding = "UTF-8"), collapse = "\n");
  teksty <- rbind(teksty, data.frame(nazwa=name, raw_text=text, processed_text=0));
}

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/AlynSmith_VertsALE.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa

```

## linia znaleziona w 'Wypowiedzi/BarbaraAnnGibson\_Renew.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/BenHabib\_NI.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/BlllNewtonDunn\_Renew.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/BillyKelleher\_Renew.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/ClaranCuffe\_VertsALE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/ClareDaly\_GUENGL.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/DanutaMariaHubner\_PPE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/DavidMcAllister\_PPE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/DianeDodds\_NI.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/DomenecRuizDevesa\_SD.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/ElleChowns\_VertsALE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/EstebanGonzalezPons\_PPE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/EstherdeLange\_PPE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/FrancesFitzgerald\_PPE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/GeoffreyVanOrden\_ECR.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/GuyVerhofstadt\_Renew.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/IlhanKyuchyuk\_Renew.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/JaakMadison\_ID.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/JamesWells\_NI.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/JanZahradil\_ECR.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/JillEvans\_VertsALE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/JudeKirtonDarling\_SD.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/JulieWard\_SD.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/KatarinaBarley\_SD.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/LanceForman\_NI.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/LudekNiedermayer\_PPE.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/LuisGaricano\_Renew.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

## linia znaleziona w 'Wypowiedzi/LukeMingFlanagan\_GUENGL.txt'

## Warning in readLines(file\_list[i], encoding = "UTF-8"): niekompletna końcowa

```
## linia znaleziona w 'Wypowiedzi/MartinaAnderson_GUENGL.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/MartinEdwardDaubney_NI.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/MiapetraKumpula-Natri_SD.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/MollyScottCato_VertsALE.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/NaoimLong_Renew.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/NicolaBeer_Renew.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/Nigel Farage_NI.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/PauloRangel_PPE.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/RadoslawSikorski_PPE.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/RichardCorbett_SD.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/SeanKelly_PPE.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/SebDance_SD.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/TerryReintke_VertsALE.txt'

## Warning in readLines(file_list[i], encoding = "UTF-8"): niekompletna końcowa
## linia znaleziona w 'Wypowiedzi/VladimirBilcik_PPE.txt'

#usuwanie części, których osoba mówiąca nie wypowiedziała, a także konwercja '\n' na ' '
```

```

for(i in 1:n) {
  teksty$processed_text[i] <- gsub("\n\\(.*?\n)", "", teksty$raw_text[i]);
  teksty$processed_text[i] <- gsub("\n\\n", " ", teksty$processed_text[i]);
  teksty$processed_text[i] <- gsub("\n", " ", teksty$processed_text[i]);
}

```

#analiza sentymentu dla każdego polityka:

#analiza sentymentu dla każdego polityka używając słownika GI

```

teksty$GI_positive <- 0
teksty$GI_negative <- 0
teksty$GI_neutral <- 0

```

```

data(DictionaryGI)
summary(DictionaryGI)

```

```

##      Length Class  Mode
## negative 2005  -none- character
## positive 1637  -none- character

```

```

for(i in 1:n) {
  sentences <- tokenize_sentences(teksty$processed_text[i]][[1]]
  sentiment <- analyzeSentiment(sentences)
  sentimentGI <- convertToDirection(sentiment$SentimentGI)

```

```

  zliczanie <- table(sentimentGI)

```

```

  teksty$GI_positive[i] <- ifelse("positive" %in% names(zliczanie), zliczanie["positive"], 0)
  teksty$GI_negative[i] <- ifelse("negative" %in% names(zliczanie), zliczanie["negative"], 0)
  teksty$GI_neutral[i] <- ifelse("neutral" %in% names(zliczanie), zliczanie["neutral"], 0)
}

```

#analiza sentymentu dla każdego polityka używając słownika HE

```

teksty$HE_positive <- 0
teksty$HE_negative <- 0
teksty$HE_neutral <- 0

```

```

data(DictionaryHE)
summary(DictionaryHE)

```

```

##      Length Class  Mode

```

```

## negative 85 -none- character
## positive 105 -none- character

for(i in 1:n) {
  sentences <- tokenize_sentences(teksty$processed_text[i])[[1]]
  sentiment <- analyzeSentiment(sentences)
  sentimentHE <- convertToDirection(sentiment$SentimentHE)

  zliczanie <- table(sentimentHE)

  teksty$HE_positive[i] <- ifelse("positive" %in% names(zliczanie), zliczanie["positive"], 0)
  teksty$HE_negative[i] <- ifelse("negative" %in% names(zliczanie), zliczanie["negative"], 0)
  teksty$HE_neutral[i] <- ifelse("neutral" %in% names(zliczanie), zliczanie["neutral"], 0)
}

#analiza sentymentu dla każdego polityka używając słownika LM
teksty$LM_positive <- 0
teksty$LM_negative <- 0
teksty$LM_neutral <- 0

data(DictionaryLM)
summary(DictionaryLM)

##      Length Class  Mode
## negative  2355 -none- character
## positive   354 -none- character
## uncertainty 297 -none- character

for(i in 1:n) {
  sentences <- tokenize_sentences(teksty$processed_text[i])[[1]]
  sentiment <- analyzeSentiment(sentences)
  sentimentLM <- convertToDirection(sentiment$SentimentLM)

  zliczanie <- table(sentimentLM)

  teksty$LM_positive[i] <- ifelse("positive" %in% names(zliczanie), zliczanie["positive"], 0)
  teksty$LM_negative[i] <- ifelse("negative" %in% names(zliczanie), zliczanie["negative"], 0)
  teksty$LM_neutral[i] <- ifelse("neutral" %in% names(zliczanie), zliczanie["neutral"], 0)
}

#analiza sentymentu dla każdego polityka używając słownika QDAP

```

```

teksty$QDAP_positive <- 0
teksty$QDAP_negative <- 0
teksty$QDAP_neutral <- 0

qdap <- loadDictionaryQDAP()
summary(qdap)

## Dictionary type: binary (positive / negative)
## Total entries: 4232
## Positive entries: 1280 (30.25%)
## Negative entries: 2952 (69.75%)

for(i in 1:n) {
  sentences <- tokenize_sentences(teksty$processed_text[i]][[1]]
  sentiment <- analyzeSentiment(sentences)
  sentimentQDAP <- convertToDirection(sentiment$SentimentQDAP)

  zliczanie <- table(sentimentQDAP)

  teksty$QDAP_positive[i] <- ifelse("positive" %in% names(zliczanie), zliczanie["positive"], 0)
  teksty$QDAP_negative[i] <- ifelse("negative" %in% names(zliczanie), zliczanie["negative"], 0)
  teksty$QDAP_neutral[i] <- ifelse("neutral" %in% names(zliczanie), zliczanie["neutral"], 0)
}

#analiza ogólnego sentymentu w poszczególnych partiach

teksty$partia <- sub(".*_(.*)$", "\\1", teksty$nazwa)
partie <- unique(teksty$partia)

teksty$partia <- as.character(teksty$partia)

m=length(partie)

#Sumowanie sentymentu poszczególnych polityków i wyrażenie tego w procentach

#Sumowanie sentymentu poszczególnych polityków i wyrażenie tego w procentach używając słownika GI

wyniki_sumaryczne <- data.frame(partie)

wyniki_sumaryczne$positive_GI <- 0
wyniki_sumaryczne$neutral_GI <- 0

```



```
wyniki_sumaryczne$negative_GI <- 0
```

```
for (k in 1:m) {  
  sum_positive <- 0  
  sum_negative <- 0  
  sum_neutral <- 0  
  for (i in 1:n) {  
    if (partie[k]==teksty$partia[i]) {  
      sum_positive <- sum_positive + teksty$GI_positive[i]  
      sum_negative <- sum_negative + teksty$GI_negative[i]  
      sum_neutral <- sum_neutral + teksty$GI_neutral[i]  
    }  
  }  
  suma <- sum_negative + sum_neutral +sum_positive  
  wyniki_sumaryczne$positive_GI[k] <- sum_positive/suma  
  wyniki_sumaryczne$neutral_GI[k] <- sum_neutral/suma  
  wyniki_sumaryczne$negative_GI[k] <- sum_negative/suma  
}
```

```
#Sumowanie sentymentu poszczególnych polityków i wyrażenie tego w procentach używając słownika HE
```

```
wyniki_sumaryczne$positive_HE <- 0  
wyniki_sumaryczne$neutral_HE <- 0  
wyniki_sumaryczne$negative_HE <- 0  
  
for (k in 1:m) {  
  sum_positive <- 0  
  sum_negative <- 0  
  sum_neutral <- 0  
  for (i in 1:n) {  
    if (partie[k]==teksty$partia[i]) {  
      sum_positive <- sum_positive + teksty$HE_positive[i]  
      sum_negative <- sum_negative + teksty$HE_negative[i]  
      sum_neutral <- sum_neutral + teksty$HE_neutral[i]  
    }  
  }  
  suma <- sum_negative + sum_neutral +sum_positive  
  wyniki_sumaryczne$positive_HE[k] <- sum_positive/suma  
  wyniki_sumaryczne$neutral_HE[k] <- sum_neutral/suma  
  wyniki_sumaryczne$negative_HE[k] <- sum_negative/suma  
}
```

#Sumowanie sentymentu poszczególnych polityków i wyrażenie tego w procentach używając słownika LM

```
wyniki_sumaryczne$positive_LM <- 0
wyniki_sumaryczne$neutral_LM <- 0
wyniki_sumaryczne$negative_LM <- 0

for (k in 1:m) {
  sum_positive <- 0
  sum_negative <- 0
  sum_neutral <- 0
  for (i in 1:n) {
    if (partie[k]==teksty$partia[i]) {
      sum_positive <- sum_positive + teksty$LM_positive[i]
      sum_negative <- sum_negative + teksty$LM_negative[i]
      sum_neutral <- sum_neutral + teksty$LM_neutral[i]
    }
  }
  suma <- sum_negative + sum_neutral +sum_positive
  wyniki_sumaryczne$positive_LM[k] <- sum_positive/suma
  wyniki_sumaryczne$neutral_LM[k] <- sum_neutral/suma
  wyniki_sumaryczne$negative_LM[k] <- sum_negative/suma
}
```

#Sumowanie sentymentu poszczególnych polityków i wyrażenie tego w procentach używając słownika

QDAP

```
wyniki_sumaryczne$positive_QDAP <- 0
wyniki_sumaryczne$neutral_QDAP <- 0
wyniki_sumaryczne$negative_QDAP <- 0

for (k in 1:m) {
  sum_positive <- 0
  sum_negative <- 0
  sum_neutral <- 0
  for (i in 1:n) {
    if (partie[k]==teksty$partia[i]) {
      sum_positive <- sum_positive + teksty$QDAP_positive[i]
      sum_negative <- sum_negative + teksty$QDAP_negative[i]
      sum_neutral <- sum_neutral + teksty$QDAP_neutral[i]
    }
  }
```

```

}
suma <- sum_negative + sum_neutral +sum_positive
wyniki_sumaryczne$positive_QDAP[k] <- sum_positive/suma
wyniki_sumaryczne$neutral_QDAP[k] <- sum_neutral/suma
wyniki_sumaryczne$negative_QDAP[k] <- sum_negative/suma
}

#Przedstawienie sentymentu sumarycznego na wykresach

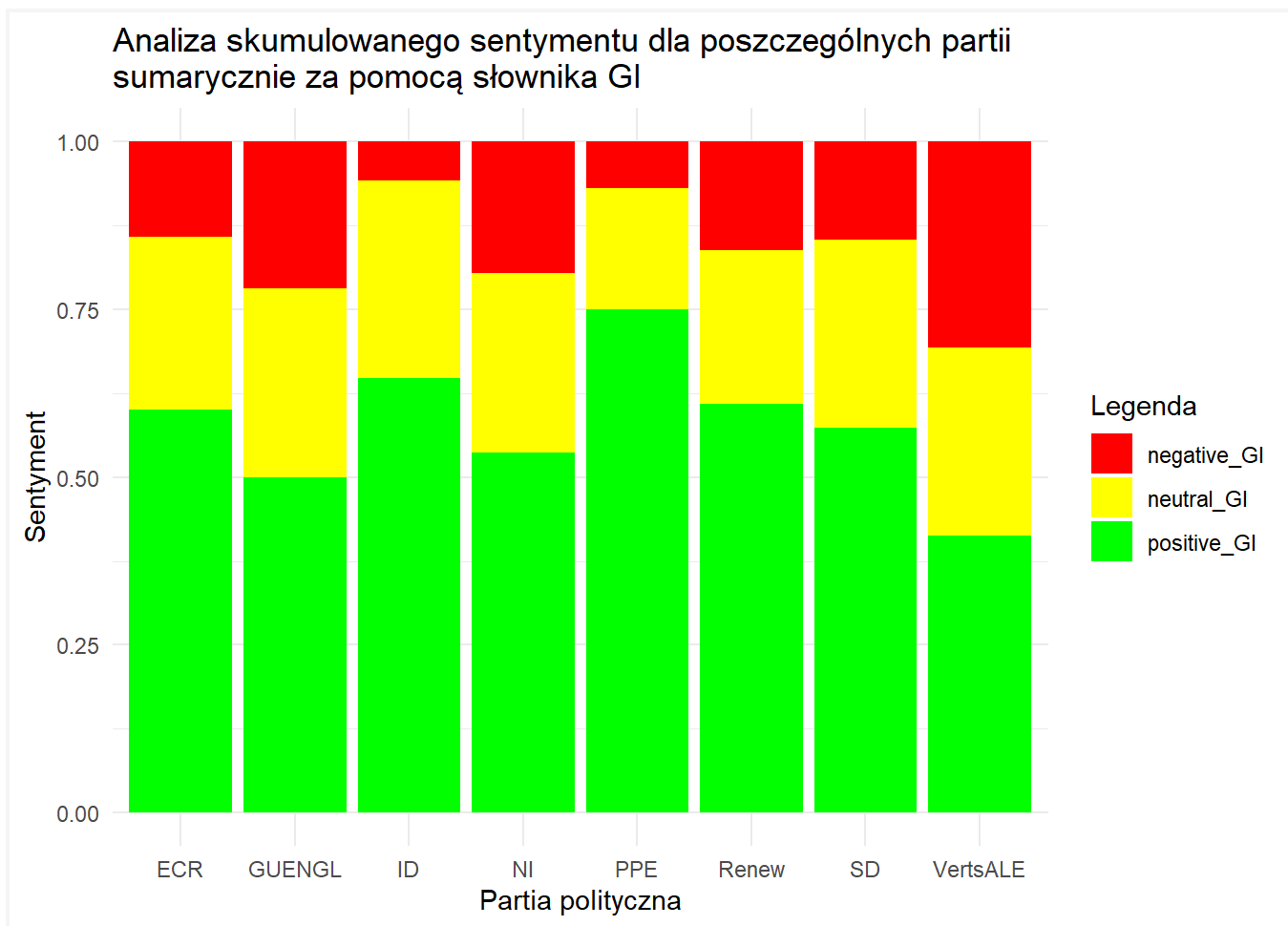
#Przedstawienie sentymentu sumarycznego według słownika GI na wykresie

df_sub_GI_sumarycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_GI, neutral_GI, negative_GI)

df_long_GI_sumarycznie <- pivot_longer(df_sub_GI_sumarycznie, cols = positive_GI:negative_GI,
names_to = "Legenda", values_to = "Value")

ggplot(df_long_GI_sumarycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\nsumarycznie za pomocą
słownika GI",
       x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_GI" = "green", "neutral_GI" = "yellow", "negative_GI" = "red")) +
  theme_minimal()

```

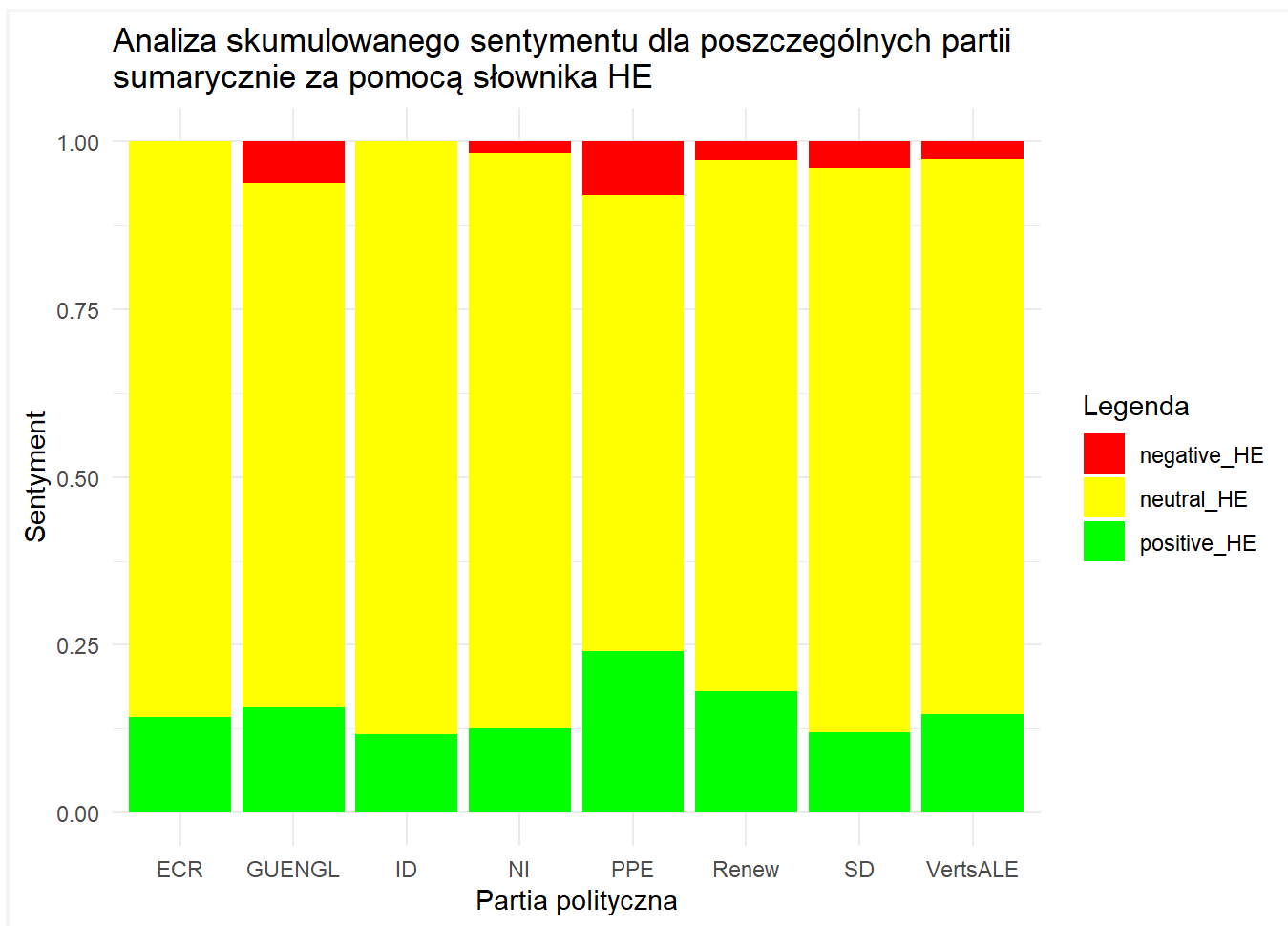


#Przedstawienie sentymentu sumarycznego według słownika HE na wykresie

```
df_sub_HE_sumarycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_HE, neutral_HE,
negative_HE)
```

```
df_long_HE_sumarycznie <- pivot_longer(df_sub_HE_sumarycznie, cols = positive_HE:negative_HE,
names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_HE_sumarycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\nsumarycznie za pomocą
słownika HE",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_HE" = "green", "neutral_HE" = "yellow", "negative_HE" = "red")) +
  theme_minimal()
```

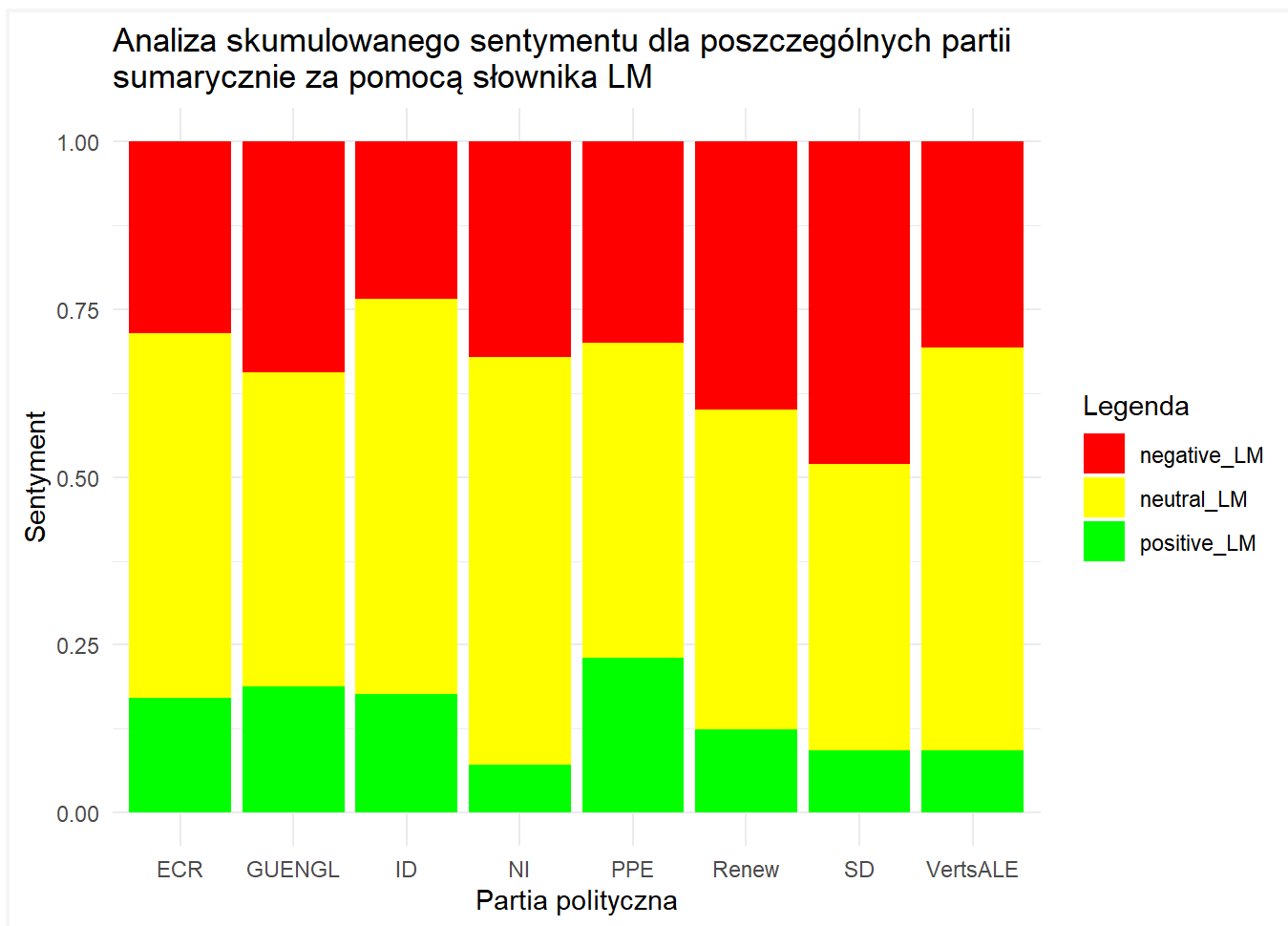


#Przedstawienie sentymentu sumarycznego według słownika LM na wykresie

```
df_sub_LM_sumarycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_LM, neutral_LM,
negative_LM)
```

```
df_long_LM_sumarycznie <- pivot_longer(df_sub_LM_sumarycznie, cols = positive_LM:negative_LM,
names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_LM_sumarycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\nsumarycznie za pomocą
słownika LM",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_LM" = "green", "neutral_LM" = "yellow", "negative_LM" = "red")) +
  theme_minimal()
```

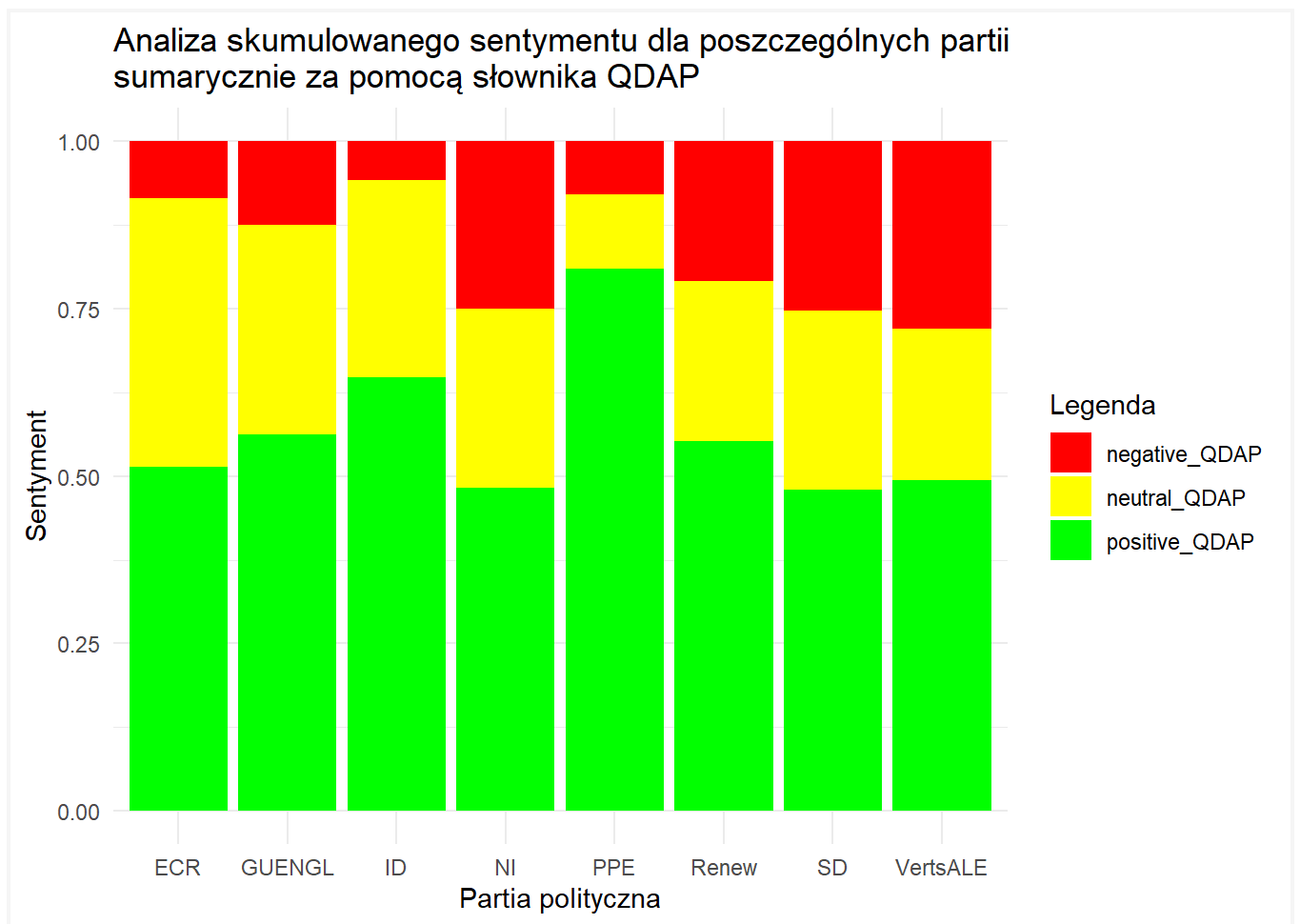


#Przedstawienie sentymentu sumarycznego według słownika QDAP na wykresie

```
df_sub_QDAP_sumarycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_QDAP, neutral_QDAP,
negative_QDAP)
```

```
df_long_QDAP_sumarycznie <- pivot_longer(df_sub_QDAP_sumarycznie, cols =
positive_QDAP:negative_QDAP, names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_QDAP_sumarycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\nsumarycznie za pomocą
słownika QDAP",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_QDAP" = "green", "neutral_QDAP" = "yellow", "negative_QDAP" =
"red")) +
  theme_minimal()
```



#Wyliczanie średnich arytmetycznych sentymentu polityków należących do każdej partii

```
wyniki_arytmetyczne <- data.frame(partie)
```

#Wyliczanie średnich arytmetycznych sentymentu polityków należących do każdej partii używając słownika  
GI

```
wyniki_arytmetyczne$positive_GI <- 0
wyniki_arytmetyczne$neutral_GI <- 0
wyniki_arytmetyczne$negative_GI <- 0
```

```
for (k in 1:m) {
  sum_positive <- 0
  sum_negative <- 0
  sum_neutral <- 0
  j <- 0;
  for (i in 1:n) {
```

```

    if (partie[k]==teksty$partia[i]) {
      suma <- teksty$GI_positive[i] + teksty$GI_negative[i] + teksty$GI_neutral[i]
      sum_positive <- sum_positive + teksty$GI_positive[i]/suma
      sum_negative <- sum_negative + teksty$GI_negative[i]/suma
      sum_neutral <- sum_neutral + teksty$GI_neutral[i]/suma
      j <- j+1;
    }
  }
  wyniki_arytmetyczne$positive_GI[k] <- sum_positive/j
  wyniki_arytmetyczne$neutral_GI[k] <- sum_neutral/j
  wyniki_arytmetyczne$negative_GI[k] <- sum_negative/j
}

```

#Wylczenie średnich arytmetycznych sentymentu polityków należących do każdej partii używając słownika HE

```

wyniki_arytmetyczne$positive_HE <- 0
wyniki_arytmetyczne$neutral_HE <- 0
wyniki_arytmetyczne$negative_HE <- 0

for (k in 1:m) {
  sum_positive <- 0
  sum_negative <- 0
  sum_neutral <- 0
  j <- 0;
  for (i in 1:n) {
    if (partie[k]==teksty$partia[i]) {
      suma <- teksty$HE_positive[i] + teksty$HE_negative[i] + teksty$HE_neutral[i]
      sum_positive <- sum_positive + teksty$HE_positive[i]/suma
      sum_negative <- sum_negative + teksty$HE_negative[i]/suma
      sum_neutral <- sum_neutral + teksty$HE_neutral[i]/suma
      j <- j+1;
    }
  }
  wyniki_arytmetyczne$positive_HE[k] <- sum_positive/j
  wyniki_arytmetyczne$neutral_HE[k] <- sum_neutral/j
  wyniki_arytmetyczne$negative_HE[k] <- sum_negative/j
}

```

#Wylczenie średnich arytmetycznych sentymentu polityków należących do każdej partii używając słownika LM



```

wyniki_arytmetyczne$positive_LM <- 0
wyniki_arytmetyczne$neutral_LM <- 0
wyniki_arytmetyczne$negative_LM <- 0

for (k in 1:m) {
  sum_positive <- 0
  sum_negative <- 0
  sum_neutral <- 0
  j <- 0;
  for (i in 1:n) {
    if (partie[k]==teksty$partia[i]) {
      suma <- teksty$LM_positive[i] + teksty$LM_negative[i] + teksty$LM_neutral[i]
      sum_positive <- sum_positive + teksty$LM_positive[i]/suma
      sum_negative <- sum_negative + teksty$LM_negative[i]/suma
      sum_neutral <- sum_neutral + teksty$LM_neutral[i]/suma
      j <- j+1;
    }
  }
  wyniki_arytmetyczne$positive_LM[k] <- sum_positive/j
  wyniki_arytmetyczne$neutral_LM[k] <- sum_neutral/j
  wyniki_arytmetyczne$negative_LM[k] <- sum_negative/j
}

```

#Wyliczenie średnich arytmetycznych sentymentu polityków należących do każdej partii używając słownika QDAP

```

wyniki_arytmetyczne$positive_QDAP <- 0
wyniki_arytmetyczne$neutral_QDAP <- 0
wyniki_arytmetyczne$negative_QDAP <- 0

for (k in 1:m) {
  sum_positive <- 0
  sum_negative <- 0
  sum_neutral <- 0
  j <- 0;
  for (i in 1:n) {
    if (partie[k]==teksty$partia[i]) {
      suma <- teksty$QDAP_positive[i] + teksty$QDAP_negative[i] + teksty$QDAP_neutral[i]
      sum_positive <- sum_positive + teksty$QDAP_positive[i]/suma
      sum_negative <- sum_negative + teksty$QDAP_negative[i]/suma
    }
  }
}

```

```

    sum_neutral <- sum_neutral + teksty$QDAP_neutral[i]/suma
  j <- j+1;
}
}
wyniki_arytmetyczne$positive_QDAP[k] <- sum_positive/j
wyniki_arytmetyczne$neutral_QDAP[k] <- sum_neutral/j
wyniki_arytmetyczne$negative_QDAP[k] <- sum_negative/j
}

```

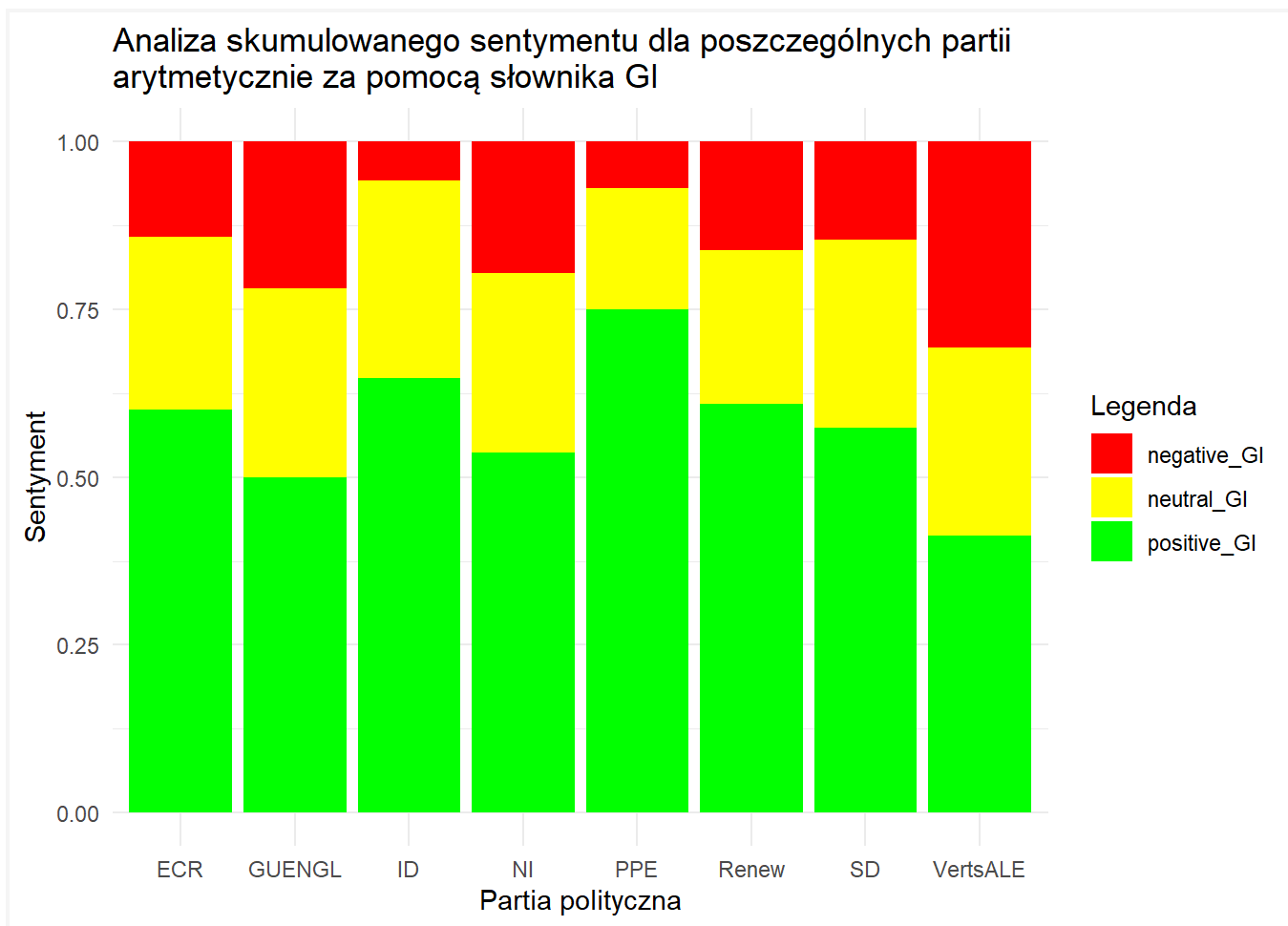
#Przedstawienie sentymentu arytmetycznego na wykresach

#Przedstawienie sentymentu arytmetycznego według słownika GI na wykresie

```
df_sub_GI_arytmetycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_GI, neutral_GI, negative_GI)
```

```
df_long_GI_arytmetycznie <- pivot_longer(df_sub_GI_arytmetycznie, cols = positive_GI:negative_GI,
names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_GI_arytmetycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\narytmetycznie za pomocą
słownika GI",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_GI" = "green", "neutral_GI" = "yellow", "negative_GI" = "red")) +
  theme_minimal()
```

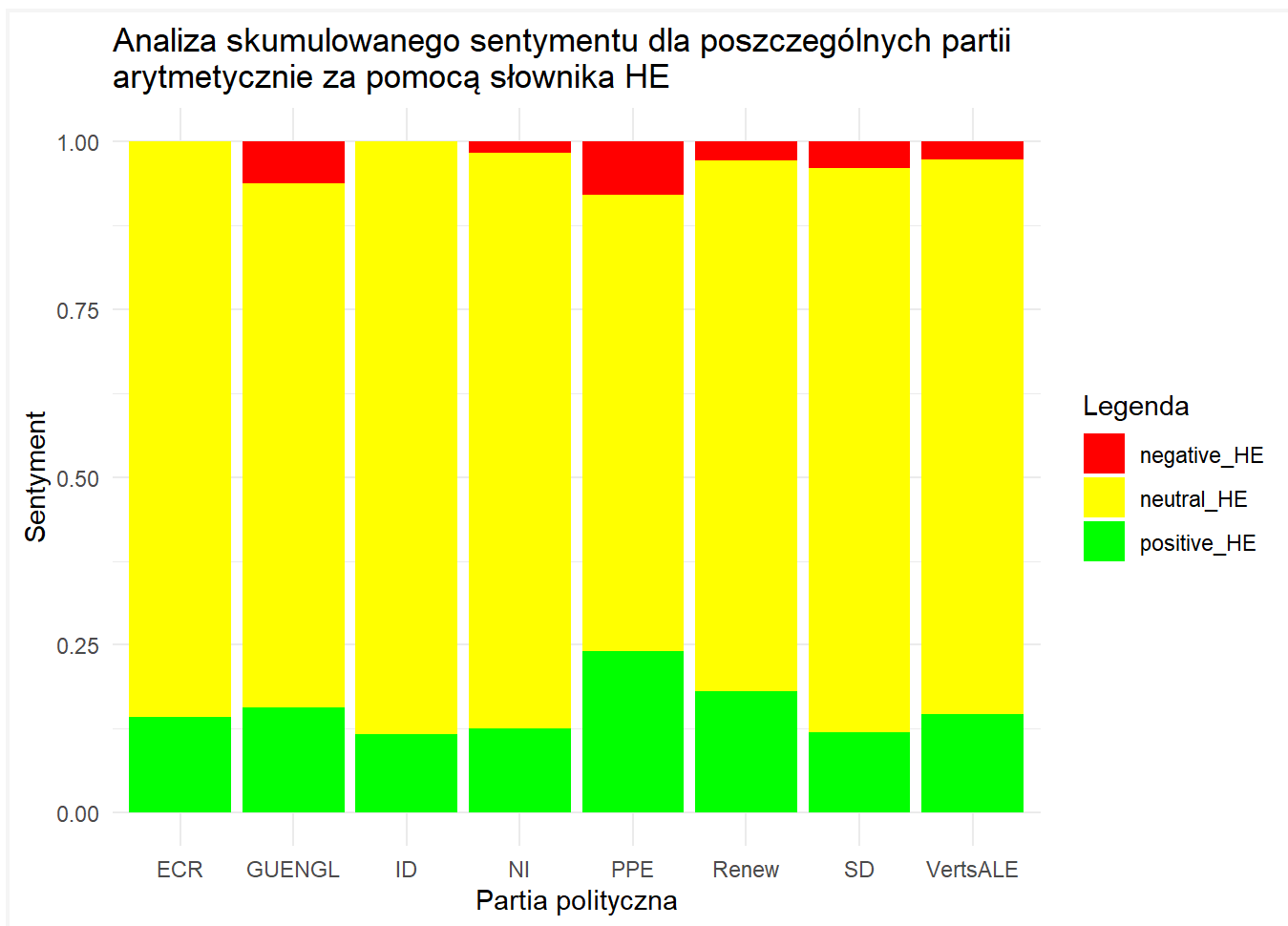


#Przedstawienie sentymentu arytmetycznego według słownika HE na wykresie

```
df_sub_HE_arytmetycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_HE, neutral_HE,
negative_HE)
```

```
df_long_HE_arytmetycznie <- pivot_longer(df_sub_HE_arytmetycznie, cols = positive_HE:negative_HE,
names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_HE_arytmetycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\narytmetycznie za pomocą
słownika HE",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_HE" = "green", "neutral_HE" = "yellow", "negative_HE" = "red")) +
  theme_minimal()
```

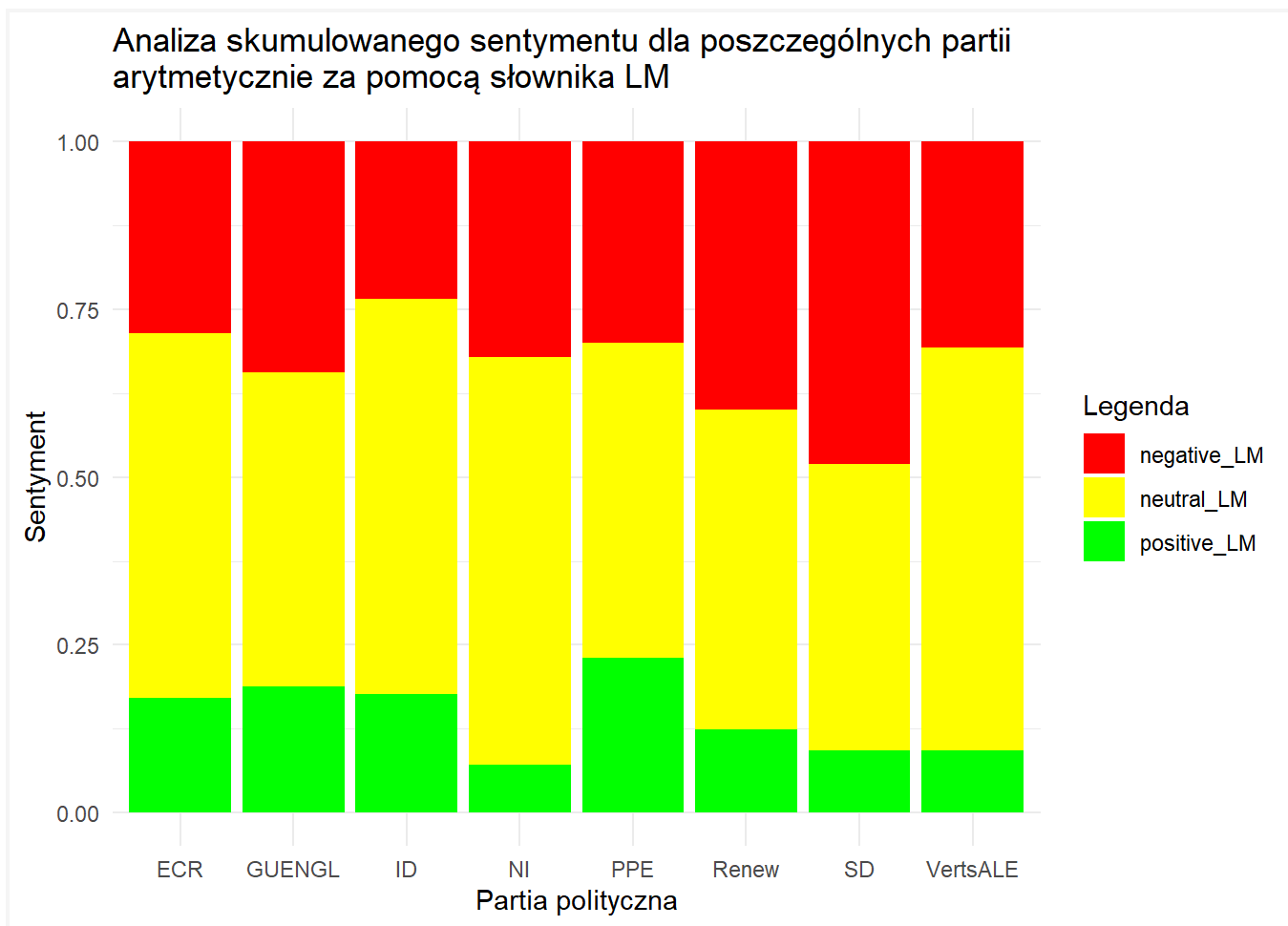


#Przedstawienie sentymentu arytmetycznego według słownika LM na wykresie

```
df_sub_LM_arytmetycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_LM, neutral_LM,
negative_LM)
```

```
df_long_LM_arytmetycznie <- pivot_longer(df_sub_LM_arytmetycznie, cols = positive_LM:negative_LM,
names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_LM_arytmetycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\narytmetycznie za pomocą
słownika LM",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_LM" = "green", "neutral_LM" = "yellow", "negative_LM" = "red")) +
  theme_minimal()
```



#Przedstawienie sentymentu arytmetycznego według słownika QDAP na wykresie

```
df_sub_QDAP_arytmetycznie <- dplyr::select(wyniki_sumaryczne, partie, positive_QDAP, neutral_QDAP,
negative_QDAP)
```

```
df_long_QDAP_arytmetycznie <- pivot_longer(df_sub_QDAP_arytmetycznie, cols =
positive_QDAP:negative_QDAP, names_to = "Legenda", values_to = "Value")
```

```
ggplot(df_long_QDAP_arytmetycznie, aes(x = partie, y = Value, fill = Legenda)) +
  geom_bar(stat = "identity") +
  labs(title = "Analiza skumulowanego sentymentu dla poszczególnych partii\narytmetycznie za pomocą
słownika QDAP",
  x = "Partia polityczna", y = "Sentyment") +
  scale_fill_manual(values = c("positive_QDAP" = "green", "neutral_QDAP" = "yellow", "negative_QDAP" =
"red")) +
  theme_minimal()
```

Analiza skumulowanego sentymentu dla poszczególnych partii  
arytmetycznie za pomocą słownika QDAP

