Sentimen analisis menggunakan R

(Pelatihan data sains menggunakan R dan Gephi)

Ujang Fahmi

Pelajaran ke-7

Salam kenal dan selamat datang.
Semoga kita semua bisa saling berbagi pengalaman dan pengetahuan. Saya adalah Ujang Fahmi, Co-founder dan mentor Sadasa Academy.

Jika anda berada dan sedang membaca tutorial ini, maka kemungkinan anda adalah orang yang sedang ingin belajar data sains, atau mungkin ditugaskan untuk mempelajari R oleh institusi atau organisasi anda. Sama seperti saya dulu, dimana tanpa latar belakang enginering saya didiharuskan untuk belajar R, demi menyelesaikan tugas akhir dan akhirnya jadilah seperti saya sekarang ini.

Satu hal yang pasti, ini adalah langkah pertama dari banyak langkah yang harus dilalui, entah melalui lembaga resmi atau belajar secara mandiri. Jadi selamat belajar!!!

Ujang Fahmi,
Yogyakarta, 2021-09-28

Materi yang disampaikan disimpan dan dokumentasikan **disini**

Apa?

Analisis sentimen adalah sebuah proses mendeteksi sentimen positif atau negataif dalam sebuah teks. Hal ini sering digunakan untuk mendeteksi data sosial, reputasi aktor/nama/brand dan memahami apa yang dibicarakan dalam teks.

Misalnya, sebuah perusahaan dapat melakukan analisis sentimen untuk mengetahui apakah konsumennya senang dengan produk/layanan atau brandnya. Data yang digunakan misalnya postingan di Twitter dengan hashtag terkait dengan perusahaannya.

Bagiamana?

Sentimen analisis bisa dilakukan dengan dua cara:

1. Lexicon Based
2. Supervised Machine Learning

Di sini, kita akan mencoba untuk membuat sentimen analisis dengan menggunakan leksikon terlebih dahulu. Hasil yang didapat kemudian bisa dijadikan sebagai basis pembuatan data latih untuk membuat supervised machine learning atau juga bisa diinterpretasikan secara langsung.

Leksikon

Leksikon pada dasarnya merupakan sebuah kamus di mana setiap term/kata memiliki sebuah value. Value untuk setiap term tersebut merupakan hasil penelitian yang umumnya dilakukan oleh akademisi dengan basis linguistik.
Di R terdapat beberapa leksikon yang bisa digunakan dari package yang ada. Misalnya dalam library `tidytext` terdapat leksikon `bing`, `afinn`, `loughran` dan `nrc`, untuk mendapatkannya bisa menggunakan skrip berikut:

```
library(tidytext)
bing_lex = get_sentiments("bing")
head(bing_lex)
```

Perbedaan antar leksikon 1

Leksikon `bing`

word	sentiment
2-faces	negative
abnormal	negative
abolish	negative
abominable	negative
abominably	negative

Leksikon `afinn`

word	value
abandon	-2
abandoned	-2
abandons	-2
abducted	-2
abduction	-2

Perbedaan antar leksikon 2

Leksikon `loughran`

word	sentiment
abandon	negative
abandoned	negative
abandoning	negative
abandonment	negative
abandonments	negative

Leksikon `nrc`

word	sentiment
abacus	trust
abandon	fear
abandon	negative
abandon	sadness
abandoned	anger

Leksinkon Bahasa Indonesia

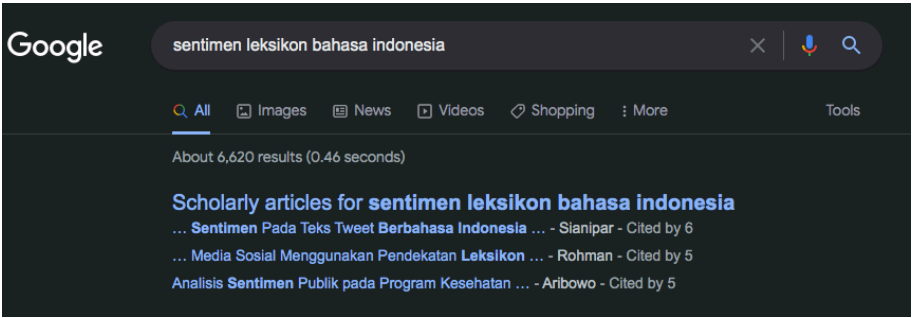

Figure 1: Leksikon sentimen bahasa Indonesia

Sama dengan bahasa Inggris, leksikon untuk analisis sentimen dalam bahasa Indonesia juga sudah banyak diteliti dengan berbagai macam metode. Kita bisa mencari artikel ilmiah dan juga kamus nya untuk kemudian digunakan yang salah satunya bisa didapat disini.

Langkah-langkah analisis sentimen

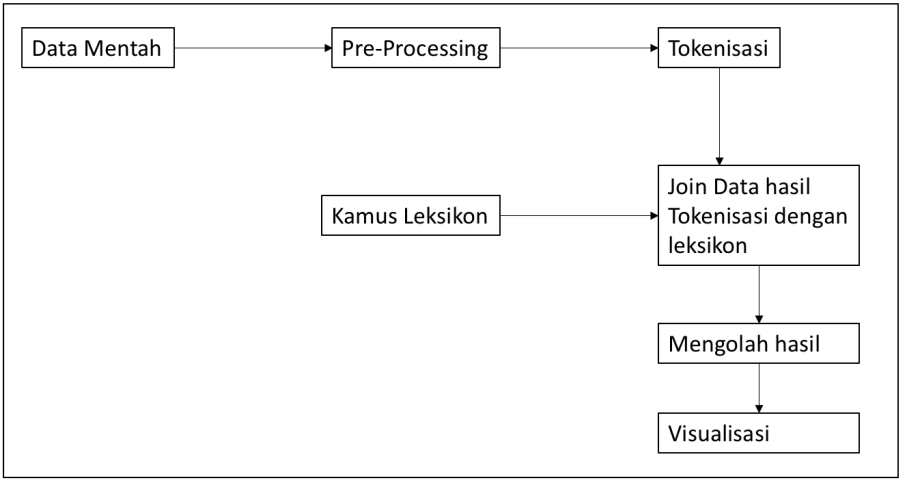

Figure 2: Proses analisis sentimen menggunakan leksikon

Penghitungan sentimen

Teks	Nilai
@DKIJakarta @aniesbaswedan : HENTIKAN Revitalisasi Monas dan kembalikan seperti semula sebagai paru paru Kota Jakarta. #SaveMonasSaveJakarta #JakartaButuhPohon - Tandatangani Petisi! https://t.co/5tjCASFzDD lewat @ChangeOrg_ID	-3

Token	Nilai
HENTIKAN	-3
Revitalisasi	0
Monas	0
kembalikan	0
semula	0
paru	0
Kota	0
Tandatangani	0

Nilai akhir sentiment sebuah teks merupakan hasil penjumlahan nilai masing-masing leksikon.

Nilai akhir kalimat/postingan diatas adalah **-3**, sehingga bisa dikategorikan Negatif.

Figure 3: Proses sentimen menggunakan leksikon

Data yang akan dianalisis

```
library(tidyverse)
library(tidytext)

raw_data = read_csv("data/tweet_save_monas.csv")
raw_data = raw_data %>%
   select(id, created_at, full_text, full_text_clean,
          reply_count, retweet_count, like_count) %>%
   filter(!is.na(full_text_clean)) %>%
   filter(!duplicated(id))
glimpse(raw_data)
```

Leksikon yang akan digunakan

Leksikon yang akan digunakan dapat diambil dari tautan berikut:

1. Leksikon Positif: `https://raw.githubusercontent.com/fajri91/InSet/master/positive.tsv`
2. Leksikon Negatif: `https://raw.githubusercontent.com/fajri91/InSet/master/negative.tsv`

```
library(tidyverse)

id_pos = read_tsv("ganti dengan tautan")
id_neg = read_tsv("ganti dengan tautan")
id_sentimen = bind_rows(id_pos, id_neg)
glimpse(id_sentimen)
```

Tokenisasi Data

```
data_token = raw_data %>%
   group_by(id) %>%
   unnest_tokens(word, full_text_clean, token = "words")

glimpse(data_token)
```

Join dengan leksikon

Setelah melakukan tokenisasi, kita bisa menggabungkan data hasil tokenisasi dengan kamus leksikon. Sehingga data yang didapatkan adalah data dengan nilai berupa numerik.

```
hasil = data_token %>%
   inner_join(id_sentimen)

hasil = hasil %>%
   group_by(id) %>%
   summarise(nilai_akhir = sum(weight))

glimpse(hasil)
```

Join dengan data asli

```
# penggabungan
hasil_akhir = raw_data %>%
   left_join(hasil)
# nilai NA menjadi 0
hasil_akhir$nilai_akhir[is.na(hasil_akhir$nilai_akhir)] = 0
# memberi label
hasil_akhir = hasil_akhir %>%
   mutate(sentimen = case_when(
      nilai_akhir == 0 ~ "Netral",
      nilai_akhir >= 1 ~ "Positif",
      TRUE ~ "Negatif"
   ))

glimpse(hasil_akhir)
View(hasil_akhir)
```

Persentase sentimen

```
persen_sentimen = hasil_akhir %>%
   count(sentimen) %>%
   mutate(persen = round(n/sum(n)*100))

library(echarts4r)
persen_sentimen %>%
   e_chart(x = sentimen) %>%
   e_pie(persen)
```

Distribusi Sentimen

```
distribusi_sentimen = hasil_akhir %>%
   separate(created_at, into = c("created_at", "jam"), sep = " ") %>%
   group_by(created_at) %>%
   count(sentimen)

distribusi_sentimen$created_at =
   as.Date(distribusi_sentimen$created_at)
glimpse(distribusi_sentimen)

distribusi_sentimen %>%
   ggplot(aes(x = created_at, y = n, color = sentimen)) +
   geom_line()
```

Table of Contents