Pre-processing untuk SNA

(Pelatihan data sains menggunakan R dan Gephi)

Ujang Fahmi

Pelajaran ke-10

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, 2021-09-30

Materi yang disampaikan disimpan dan dokumentasikan **disini**

Social Network Analyss

Sama halnya dengan pengeolahan data lain, untuk bisa melakukan SNA kita juga perlu melakukan persiapan terlebih dahulu. Persiapan ini meliputi:

1. Data yang akan dianalisis;
2. Tujuan/pertanyaan analisis;
3. Nodes dan Edges;
4. Parameter yang akan digunakan; dan
5. Tools yang akan digunakan

Data yang akan dianalisis

Data yang bisa dianalisis menggunakan SNA sebenarnya cukup beragam dan hampir semua jenis data yang didalamnya bisa kita definisikan nodes dan edgesnya bisa dibuat menjadi network. Tapi umumnya data yang akan dianalisis menggunakan SNA masih berupa data mentah dalam format csv atau graph.

Impor data

1. Impor Tabel: Untuk mengimpor data berupa tabel, misalnya dengan ekstensi xlsx, csv atau tsv kita bisa menggunakan fungsi-fungsi seperti `read_csv`, `read_excel`, atau `read_tsv` yang semuanya bisa digunakan dengan memanggil `library(tidyverse)`.
2. Impor Data Graph: Terkadang kita juga mendapatkan data graph dari aplikasi lain. Jika data/file dengan ekstensi graph seperti graphml, gexf, atau pajek akan diolah di R kita bisa menggunakan fungsi-fungsi impor dari igraph seperti `read_graph`. Informasi lebih detil: `?read_graph()`.
3. Impor Json: Untuk beberapa kasus, misalnya network yang sudah divisualkan dalam sebuah website umumnya menyimpan file dalam format json. Di R kita bisa menggunakan library jsonlite untuk mengimpor file tersebut. Informasi lebih detil: `?jsonlite`.

Wrangling dan Cleansing

Sebagai contoh, berikut adalah data yang akan dianalisis.

```
library(tidyverse)
raw_data = read_csv("data/tweet_save_monas.csv")
glimpse(raw_data)
```

Data di atas terdiri dari 3000 dan 42 variabel. Semua baris akan dianalisis, tapi untuk SNA, kita tidak memerlukan semua variabel. Untuk SNA kita hanya perlu variabel spesifik yang sesuai dengan SNA yang akan dibuat.

Memilih variabel yang akan dianalisis

Dari data tersebut, lalu kita memutuskan untuk membuat network username dengan edgese mention, dimana:

1. Nodes = Username (ada di kolom user_name dan full_text)
2. Edges = mention (Username dari kolom user_name mention username di kolom full_text)

```
raw_data = raw_data %>%
   select(id, user_name, full_text)
glimpse(raw_data)
```

Mengekstrak nodes

Sampai tahap sebelumnya, kita sudah memiliki data yang fokus untuk SNA. Tapi data tersebut juga memiliki ID, yang jika dibutuhkan bisa digabungkan kembali dengan data awalnya.

Tantangan selanjutnya adalah mengekstrak nodes atau dalam konteks ini username dari variabel kedua, yaitu full_text. Untuk melakukannya kita bisa menggunakan `regex`.

```
# involved
user_inv <- as.character(raw_data$full_text)
user_inv <- sapply(str_extract_all(user_inv, "(@[[:alnum:]_]*)",
                                   simplify = FALSE),
                   paste,
                   collapse = ", ")
user_inv <- data_frame(user_inv)
raw_data$user_mention = user_inv$user_inv
glimpse(raw_data)
```

Tujuan/pertanyaan analisis

Tujuan analisis ini bisa diawali dengan menentukan objek/nodes yang akan diteliti. Misalnya, untuk analisis media sosial nodes yang akan digunakan adalah username, sementara edgesnya adalah mention.

Sementara untuk analisis konten nodes yang bisa digunakan misalnya kata/term dan edgesnya adalah lokasi pada observasi/baris/kalimat yang sama.

Nodes dan Edges

Dalam setiap SNA, pada dasarnya adalah kita menganalisis nodes dan edges. Di mana dalam bentuk paling sederhananya dapat dilihat seperti tabel berikut.

sumber	target
n1	n3
n2	n1
n3	n6

Tabel dengan dua kolom yang masing-masingnya diisi dengan value berupa nodes seperti di atas sudah bisa dijadikan network. Di mana masing-masing nodes di kolom `sumber` terhubung oleh sebuah garis penghubung abstrak yang disebut edges dengan nodes di kolom `target`.

Parameter dalam SNA yang akan digunakan

1. Untuk menentukan parameter apa yang cocok untuk digunakan dalam analisis, kita perlu mengetahui tujuan yang ingin dicapai atau pertanyaan yang ingin dijawab.
2. Dari tujuan/pertanyaan tersebut selanjutnya kita bisa memilih parameter. Misalnya betweeness centrality, karena kita ingin mengetahui nodes yang memiliki kemungkinan bisa menjadi jembatan tersebarnya informasi.
3. Jika tujuannya adalah mengetahui komunitas, maka centrality tidak bisa digunakan melainkan modularity.
4. Jadi, dalam SNA kita tidak perlu menggunakan semua parameter/pengukuran yang ada, melainkan yang sesuai dengan tujuan analisis saja.

Tokenisasi nodes

Berdasarkan proses terakhir yang sudah dilakukan, yaitu pengekstrakan nodes di kolom `full_text`, kini kita perlu menjadikannya token, agar bisa dianggap sebagai network.

```
library(tidytext)

raw_data = raw_data %>%
   unnest_tokens(target, user_mention,
                 token = "words",
                 to_lower = FALSE)
glimpse(raw_data)
View(raw_data)
```

Memastikan data Network

Dalam kondisi real, tidak semua twit memention usrename lain dalam postingannya. Karena dalam konsep newtork satu nodes harus terhubung dengan nodes lain, kita perlu pastikan variabel yang ada masing-masing memiliki pasangan.

```
node_edge1 = raw_data %>%
   select(sumber = user_name, target)

node_edge1$sumber = paste0("@", node_edge1$sumber)
node_edge1$target = paste0("@", node_edge1$target)

View(node_edge1)
glimpse(node_edge1)
```

Tools yang akan digunakan

Tools yang akan digunakan untuk membuat analisis juga menjadi salah satu yang harus dipertimbangkan. Dalam konteks ini, misalnya kita akan menggunakan gephi atau node xl, maka setidaknya kita perlu menyediakan data yang merepresentasikan network seperti tabel dibagian nodes dan edges sebelumnya.

Selain menggunakan R, kita juga bisa menggunakan tools (tanpa koding) untuk membuat SNA: Pilihannya diantaranya adalah:

1. Nodexl (sejauh ini hanya bisa diinstall di windows dengan versi gratis dan berbayar)
2. Gephi (bisa diinstall di semua OS, gratis)

Mengkespor Data Graph

Jika kita akan mengerjakan SNA dengan software lain, maka kita perlu mengekspor data untuk SNA dalam format yang sesuai. Sebagai contoh, untuk Gephi kita bisa menggunakan format file `graphml`.

```
library(igraph)

# Membuat file graph
g1 = graph_from_data_frame(d = node_edge1,
                           directed = FALSE)
class(g1)

# menyimpan file
igraph::write_graph(graph = g1,
                    file = "data/tes_net.graphml",
                    format = "graphml")
igraph::write_graph(graph = g1,
                    file = "data/tes_net.txt",
                    format = "edgelist")
```

Cek data graph

```
nodes = data_frame(nodes = V(graph = g1)$name)
edges = get.edgelist(g1)

View(edges)
```

Table of Contents

Social Network Analyss

Data yang akan dianalisis

- Impor data
- Wrangling dan Cleansing
- Memilih variabel yang akan dianalisis
- Mengekstrak nodes

Tujuan/pertanyaan analisis

Nodes dan Edges

Parameter dalam SNA yang akan digunakan

- Tokenisasi nodes
- Memastikan data Network

Tools yang akan digunakan

- Mengkespor Data Graph
- Cek data graph