

Analisis Jejaring Sosial

Dian Ramadhani

08/01/2020

Analisis Jejaring Sosial

Analisis jejaring sosial memandang hubungan sosial sebagai simpul dan ikatan. Simpul adalah aktor individu di dalam jaringan, sedangkan ikatan adalah hubungan antar aktor tersebut.

Install Packages

```
# Menginstall package(s)
install.packages("readr") # membaca file
install.packages("igraph") # membentuk jaringan
install.packages("here") # set direktori
```

Import Library

Setelah paket diinstal, paket tersebut tidak secara otomatis aktif. Dengan demikian, paket yang telah diinstal selanjutnya akan diaktifkan melalui perintah berikut.

```
# Mengaktifkan package(s)
library(readr)
library(igraph)
library(here)
```

Set Directory

```
# Mengetahui direktori proyek
here()
```

Import Data

Langkah pertama yang dilakukan setelah mempersiapkan paket yaitu mengimpor data yang akan digunakan. Data yang akan diimpor dicari dengan menggunakan lokasi here("folder", "subfolder", "file.csv")

```
# Mengimport data
df.tweets <- read_csv(here("data", "temporary", "hasilcrawling-keyword.csv"))
```

Eksplorasi Data

Data yang telah diimpor selanjutnya dieksplorasi untuk mengetahui strukturnya.

```
# Melihat attribute dan struktur data
names(df.tweets)
dim(df.tweets)
head(df.tweets)
str(df.tweets)
```

Memilih Data

```
# Memilih data yang akan digunakan
df.select <- subset(df.tweets, select = c(screen_name, reply_to_screen_name))
```

Membuang Tweets yang Tidak Mengandung Interaksi dan Membuat Edge List

```
# Mengetahui jumlah data kosong
sum(is.na(df.select))

# Menghilangkan baris dengan data kosong (membuat dataframe edgelist)
df.edgelist <- na.omit(df.select)
```

Memodelkan Jejaring Sosial

```
# Membuat model melalui dataframe edgelist
graph <- graph_from_data_frame(df.edgelist, directed = TRUE)
graph
```

Melihat Daftar Nodes dan Edges

```
# Melihat daftar nodes
nodes <- as_data_frame(graph, what = "vertices")
View(nodes)

# Melihat daftar edges
edges <- as_data_frame(graph, what = "edges")
View(edges)
```

Visualisasi Jejaring Sosial

```
# Visualisasi jejaring sosial 1
plot(graph,
  layout = layout_fruchterman_reingold, rescale = FALSE,
  vertex.label.color = "black",
  vertex.label.size = 0, vertex.size = degree(graph) * .25,
  edge.arrow.size = .4, edge.curved = .1,
  edge.color = "orange", vertex.color = "orange",
  ylim = c(-10, 40), xlim = c(2, 30), main = "Tweets About Banjir"
)

# Visualisasi jejaring sosial 2
tkplot(graph,
  layout = layout_fruchterman_reingold, rescale = FALSE,
  vertex.label.color = "black",
  vertex.label.size = degree(graph) * .25, vertex.size = degree(graph) * .25,
  edge.arrow.size = .4, edge.curved = .1,
  edge.color = "orange", vertex.color = "orange",
  ylim = c(-10, 40), xlim = c(2, 30), main = "Tweets About Banjir"
)
```

Sentralitas pada Jaringan

```
# Degree centrality
degree.centrality <- data.frame(degree(graph,
  v = V(graph),
  mode = "all", normalized = FALSE
))
View(degree.centrality)

# Betweenness centrality
betweenness.centrality <- data.frame(betweenness(graph,
  v = V(graph),
  directed = TRUE, normalized = TRUE
))
View(betweenness.centrality)

# Closeness centrality
closeness.centrality <- data.frame(closeness(graph,
  vids = V(graph),
  mode = "all", normalized = TRUE
))
View(closeness.centrality)

# Eigenvector centrality
eigen.centrality <- data.frame(eigen_centrality(graph))
View(eigen.centrality)
```

Struktur Jaringan

```
# Diameter
diameter(graph, directed = TRUE)

# Average path length
average.path.length(graph, directed = TRUE)

# Density
edge_density(graph)

# Modularity
modularity(graph, membership(cluster_walktrap(graph)))
```