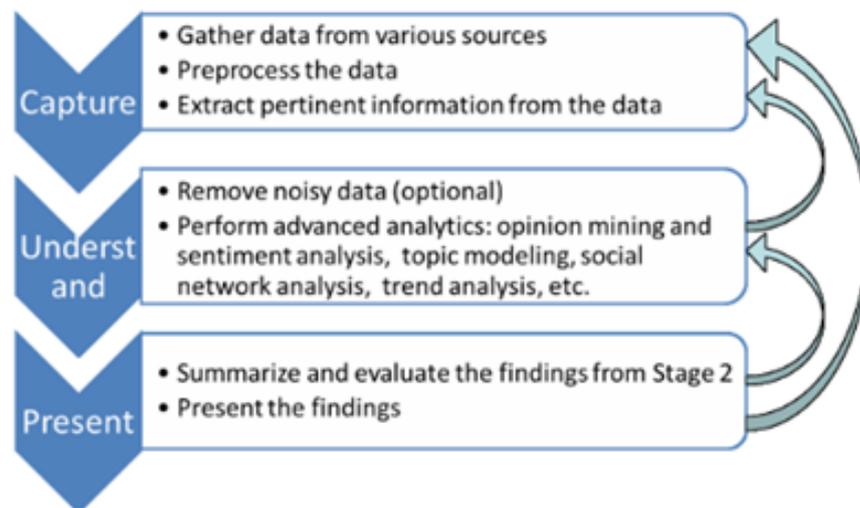


BAB 2

LANDASAN TEORI

2.1 *Social Media Analytics Process*

Social media analytics process adalah sebuah proses dalam mengumpulkan, menganalisis, dan juga memvisualisasikan data dari sosial media untuk mendapatkan sebuah pola dan pengetahuan yang berguna [5]. *Social Media Analytics* memiliki tiga tahapan proses seperti gambar dibawah ini:



Gambar 2.1 *Social Media Analytics Process*

Berikut adalah penjelasan dari setiap tahapan *Social media analytics process*:

1. *Capture*

Pada tahapan ini dilakukan proses pengumpulan dan praprosesan data sehingga data siap diolah pada langkah berikutnya. Data yang dikumpulkan pada proses ini didapatkan dari berbagai sumber, seperti crawling, maupun mengambil data sosial media yang sudah diolah sebelumnya. Praprosesan data dilakukan untuk mempersiapkan data agar siap digunakan pada tahap selanjutnya, yang meliputi seleksi data berdasarkan kebutuhan sehingga data akan menjadi lebih rapi serta terstruktur.

2. *Understand*

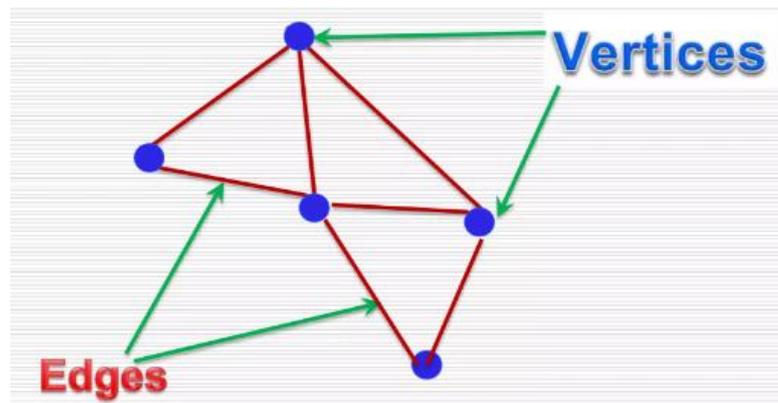
Pada tahap ini, model analisis diterapkan pada data yang telah disiapkan sebelumnya yakni *Capture*. Model yang digunakan dapat dianalisis menggunakan metode seperti analisis sentimen, analisis jejaring sosial, analisis statistik, dan metode lainnya.

3. *Present*

Pada tahap ini, dilakukan proses penyimpulan dan evaluasi hasil yang diperoleh dari pemodelan data pada tahap sebelumnya. Informasi dari kesimpulan dan evaluasi tersebut kemudian dapat disampaikan dalam bentuk visualisasi agar lebih mudah dipahami oleh orang lain.

2.2 *Social Network Analysis*

Social Network Analysis (SNA) adalah pendekatan penelitian dalam ilmu sosial dan perilaku yang berfokus pada cara-cara interaksi dan interkoneksi antara individu dan kelompok sosial untuk menjelaskan pola-pola sosial berdasarkan dari perasaan, pikiran, maupun perilaku [6]. Individu atau orang digambarkan sebagai *nodes* atau titik, sedangkan *edges* atau links menggambarkan relasi antar individu. Implementasi *Social Network Analysis* mampu memvisualisasikan relasi atau hubungan antar aktor dalam bentuk graf. Relasi dalam *Social Network Analysis* dapat dianalisis melalui perhitungan yang menghasilkan centrality dalam sebuah jaringan sosial, sesuai dengan posisi masing-masing aktor dalam struktur jaringan tersebut. Berikut adalah contoh visualisasi graf sederhana dalam *Social Network Analysis* pada Gambar 2.2: Visualisasi Graf Sederhana dalam *Social Network Analysis*



Gambar 2.2 Contoh Visualisasi Graf Sederhana *Social Network Analysis*

2.3 Centrality

Centrality adalah ukuran atau metrik yang ada pada *graph* dan *network theory*. Centrality digunakan untuk menilai relevansi (keterhubungan), atau *structural importance* dari *node* pada suatu jaringan. Pada OSN (*Online Social Network*) perhitungan centrality dapat digunakan untuk mendeteksi *the most influential people* pada suatu jaringan [3]. Dalam SNA, terdapat empat jenis centrality, yaitu *degree centrality*, *betweenness centrality*, *closeness centrality*, dan *eigenvector centrality*.

1. Degree Centrality

Degree Centrality dihitung dengan cara mengukur jumlah langsung koneksi yang dimiliki oleh sebuah *node*. *Node* dengan *Degree Centrality* tinggi dianggap lebih penting karena memiliki banyak hubungan langsung dengan *node* lain dalam jaringan [7]. Berikut adalah rumus dari *Degree Centrality* yang telah dinormalisasikan.

$$CD(n_i) = \frac{d(n_i)}{n - 1} \quad (2.1)$$

Keterangan:

$d(n_i)$ adalah links atau *edges* yang ada pada suatu jaringan graf *Social Network Analysis*.

2. *Betweenness centrality*

Betweenness centrality adalah ukuran tingkat kepentingan seseorang dalam jejaring sosial berdasarkan kemampuannya untuk menghubungkan banyak individu lain. Ukuran ini menunjukkan seberapa sering suatu *node* (individu) berada pada *shortest path* antara dua *node* lainnya dalam jaringan. Berikut adalah rumus dari *Betweenness centrality*.

$$CB(ni) = \sum \frac{\sigma(u, v|ni)}{\sigma(u, v)} \quad (2.2)$$

Keterangan:

$\sigma(u, v|ni)$ adalah banyaknya *shortest path* dari titik u ke titik v yang melewati *node* ni.

$\sigma(u, v)$ adalah jumlah *shortest path* antara titik u dan v.

3. *Closeness Centrality*

Closeness centrality diartikan sebagai ukuran tingkat kepentingan seseorang berdasarkan jarak rata-rata ke seluruh individu lain dalam jejaring sosial. *Closeness centrality* mengindikasikan seberapa cepat seseorang dapat mencapai semua orang lain dalam jaringan tersebut. Berikut adalah rumus dari *Closeness Centrality*.

$$CC(ni) = \frac{n - 1}{\sum d(ni, nj)} \quad (2.3)$$

Keterangan:

N adalah jumlah keseluruhan *edges* yang ada pada graf.

$d(ni, nj)$ adalah banyaknya *shortest path* antara titik ni dan nj.

4. *Eigenvector centrality*

Eigenvector centrality adalah ukuran tingkat kepentingan seseorang dalam jaringan sosial yang didasarkan pada tingkat kepentingan orang-orang yang terhubung dengannya [8]. Dengan kata lain, seseorang dianggap lebih penting jika dia terhubung dengan orang-orang lain yang juga penting dalam jaringan tersebut.

$$Ax = \lambda x \quad (2.4)$$

Keterangan:

A adalah *matrix adjacency* yang merepresentasikan hubungan antara simpul pada jaringan, λ adalah *eigenvalue* yang sesuai dengan *eigenvector* x rumus dari nilai *eigenvector centrality* dapat dihitung dengan menggunakan menggunakan metode perhitungan iteratif.

2.4 *Modularity*

Modularity merupakan salah satu pengukuran struktur komunitas yang sering digunakan pada jaringan sosial. Hasil dari *modularity* digunakan untuk melihat kualitas struktur jaringan dan juga untuk menentukan berapa banyak jumlah komunitas yang dibutuhkan sehingga memperoleh struktur yang paling optimal. Berikut ini adalah rumus dari perhitungan *modularity* [9].

$$M = \sum_{c=1}^{n_c} \left[\frac{L_c}{L} - \left(\frac{k_c}{2L} \right)^2 \right] \quad (2.5)$$

Modularity (M) adalah ukuran yang digunakan dalam analisis jaringan untuk mengevaluasi seberapa baik partisi jaringan telah dilakukan ke dalam komunitas-komunitas yang saling terkait. Rumusnya adalah jumlahkan selisih antara proporsi jumlah edge yang berada dalam komunitas terhadap total jumlah edge dengan proporsi kuadrat dari total derajat *node* dalam komunitas terhadap dua kali total

jumlah edge, untuk setiap komunitas dalam jaringan. Karakteristik utama *modularity* adalah sebagai berikut:

1. Tingkat *modularity* yang tinggi menandakan partisi yang lebih optimal. Semakin tinggi nilai *modularity*, semakin optimal struktur komunitas yang terbentuk. Nilai *modularity* berkisar antara 0 hingga 1. Nilai $M=1$ menandakan bahwa komunitas-komunitas dalam jaringan sangat terpisah dengan jelas, tanpa tumpang tindih, dan setiap simpul telah jelas menjadi bagian dari suatu komunitas.
2. Nilai *modularity* yang nol mengindikasikan tidak ada partisi komunitas yang terjadi, sehingga seluruh jaringan dianggap sebagai satu komunitas tunggal. Sebaliknya, jika nilai *modularity* negatif, maka masing-masing *node* dipartisi ke dalam komunitas yang berbeda secara acak.

2.5 *Community detection*

Community detection, adalah salah satu langkah dalam analisis jaringan sosial. *Community detection* bekerja dengan cara mengelompokkan simpul-simpul dalam jaringan ke dalam beberapa kelompok (komunitas) berdasarkan kesamaan dalam berbagai karakteristik, seperti minat, hobi, tujuan, gaya hidup, lokasi, atau profesi. Tujuan dari deteksi komunitas adalah untuk mengidentifikasi struktur internal dalam jaringan yang mengungkapkan pola hubungan yang signifikan di antara simpul-simpulnya .

2.6 *Louvain Community Detection Algorithm*

Louvain Community detection adalah suatu algoritma yang digunakan untuk mendeteksi komunitas. Kelebihan dalam menggunakan Louvain adalah mendeteksi komunitas dengan memperhitungkan *modularity* yang maksimal dan cenderung lebih cepat dibandingkan dengan algoritma *Community detection* lain karena pada dasarnya algoritma ini menggunakan metode *brute force* [10]. Algoritma Louvain memiliki dua fase[11]. Pada fase pertama, setiap simpul dalam jaringan dianggap sebagai komunitasnya sendiri. Kemudian, untuk setiap simpul, algoritma

menghitung perubahan nilai *modularity* saat sebuah *node* dipindahkan dari komunitasnya sendiri ke setiap komunitas tetangga. Setelah perhitungan nilai *modularity* dilakukan untuk setiap kemungkinan perpindahan, *node* tersebut ditempatkan pada komunitas yang memberikan peningkatan *modularity* terbesar. Jika tidak ada peningkatan yang signifikan, *node* tetap berada dalam komunitas asalnya. Proses ini diulang untuk setiap simpul sampai tidak ada peningkatan *modularity* yang signifikan. Pada fase kedua, jaringan baru dibangun di mana banyak *nodes* dapat mewakili komunitas-komunitas yang ditemukan selama fase pertama.

2.7 Python

Python adalah bahasa pemrograman komputer yang dirancang untuk menekankan keterbacaan dan kemudahan dalam penulisan sebuah kode. Python juga dikenal sebagai bahasa pemrograman yang memiliki sintaks sederhana dan jelas sehingga memudahkan programmer untuk menulis dan membaca suatu kode[12]. Python juga merupakan sebuah lingkungan interaktif yang memungkinkan komputasi teknis menjadi lebih produktif karena python memiliki banyak alat-alat untuk eksplorasi data yang interaktif serta perkembangan skrip yang cepat sehingga dapat berguna bagi para peneliti dan ilmuwan. Bahasa pemrograman ini dilengkapi dengan banyak *library* serta fungsi banyak didalamnya sehingga memungkinkan pengguna untuk melakukan berbagai tugas dengan efisien. Pada penelitian ini akan digunakan 2 *library* fungsi yang akan membantu didalam proses membentuk suatu jaringan *social network* yakni:

A. Python Regular Expressions (Regex)

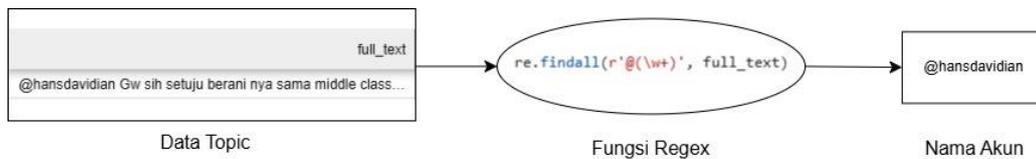
Regular expressions (regex) adalah alat penting dalam bahasa pemrograman yang digunakan untuk pencocokan pola yang ada pada data [13]. Salah satu dari fungsi regex yakni adalah 're.findall()' yang digunakan untuk menemukan semua kemunculan pola dalam data *string* dan mengembalikannya dalam bentuk *list*. Cara kerja dari fungsi 're.findall()' adalah mencari pola dalam string dan mengembalikan semua kemunculan pola tersebut dalam bentuk list. Jika tidak ada pola yang cocok

ditemukan, fungsi ini mengembalikan list kosong. Berikut adalah gambar dari fungsi regex yang bisa dilihat pada Gambar 2.3

```
re.findall(r'@(\w+)', full_text)
```

Gambar 2. 3 Fungsi Regex

Berikut adalah gambaran cara kerja Regex pada penelitian ini yang ada pada Gambar dibawah ini



Gambar 2. 4 Cara Kerja Regex

Gambar di atas adalah cara kerja Regex yang akan digunakan di penelitian ini untuk membentuk struktur data baru dengan format graf berdasarkan dari fitur mention pada X.

B. *Max Connected Components (NetworkX)*

Max connected components adalah konsep dalam analisis jaringan untuk mengidentifikasi komponen terhubung terbesar dalam sebuah graf. Analisis komponen terhubung merupakan suatu metode yang penting dilakukan untuk memahami struktur dan dinamika dari jaringan sosial [14]. NetworkX adalah *library* yang digunakan untuk pembuatan dari sebuah graf pada bahasa pemrograman python. NetworkX menyediakan fungsi untuk mengidentifikasi graf dengan komponen terhubung terbesar dengan cara menggunakan fungsi ‘connected_components()’ untuk menemukan semua komponen terhubung yang ada pada graf, lalu dilanjutkan dengan menggunakan fungsi ‘max()’ dengan parameter ‘key=len’ untuk menemukan komponen terhubung terbesar berdasarkan jumlah *node* yang ada pada komponen tersebut. Fungsi Max Connected Components dapat dilihat pada gambar 2.4

```
max(nx.connected_components(G), key=len)
```

Gambar 2. 5 Fungsi Max Connected Components

2.8 *Giant Component Graph*

Giant Component Graph adalah komponen terhubung terbesar dalam sebuah jaringan dimana komponen ini terdiri dari simpul (*node*) dan tepi (*edge*) yang membentuk subgraf di mana setiap simpul dapat mencapai simpul lainnya melalui jalur tertentu [15]. Istilah "giant component" sering digunakan dalam studi graf dan teori jaringan untuk menggambarkan bagian terbesar dan paling terhubung dari jaringan yang berisi sebagian besar *node*. Giant component membantu memahami struktur inti dari jaringan.

2.9 X (X)

X (X) merupakan sebuah platform online yang dikelola oleh X Corp., X menawarkan layanan jejaring sosial dan *microblogging* dimana penggunaanya bisa mengirim serta membaca pesan singkat yang dikenal sebagai tweet. Isi teks dalam satu tweet terbatas hingga 280 karakter, muncul di profil pengguna dan bisa diakses oleh umum, meskipun ada opsi untuk membuat pesan-pesan tersebut hanya tersedia bagi lingkaran pertemanan. Pengguna juga bisa mengikuti akun lain untuk menerima update tweet mereka, yang membuat mereka menjadi pengikut.

2.10 *Buzzer X (X)*

Buzzer X (X) adalah pengguna akun X yang dapat menyebarkan informasi secara luas. *Buzzer* secara harfiah berarti perangkat yang mengeluarkan suara keras untuk menarik perhatian. Analogi *Buzzer X (X)* mengacu pada akun yang berdampak besar, aktif men-tweet konten sehingga tweet mereka sering di-retweet, dan aktif berinteraksi dengan orang lain di media sosial X [16]. *Buzzer* adalah akun media sosial nyata atau anonim yang membagikan konten kepada publik dari media sosial dengan tujuan untuk membuat topik pada konten yang dibagikan menjadi viral.