

Social Network Analysis Using Betweenness Centrality to find Influential Users and Detect Community in Social Networks

1st Hovely Simatupang
Telkom University
Informatics Department
Bandung, Indonesia
hovelywahyu@student.telkomuniversity.ac.id

2nd Buala Leonardo Hulu
Telkom University
Informatics Department
Bandung, Indonesia
bualaleo@student.telkomuniversity.ac.id

Abstrak—

Media adalah salah satu bentuk kebutuhan yang selalu dikonsumsi oleh setiap individu. Media digunakan dengan berbagai macam tujuan dan motif yang berbeda-beda. Dalam Analisis Jaringan Sosial (SNA), sebuah informasi jejaring sosial terdiri dari 2 bagian yang dapat diteliti yaitu influential user dan community detection. Pada penelitian ini influential user diidentifikasi menggunakan algoritma betweenness centrality sedangkan untuk community detection untuk mengetahui struktur komunitas yaitu fitur penting dari jaringan yang kompleks. Deteksi komunitas pada penelitian ini memerlukan pengetahuan global tentang analisis jejaring sosial yang tidak realistis. Karena meningkatnya jaringan sosial online, tantangan baru adalah mengembangkan metode untuk mendeteksi komunitas berdasarkan informasi yang diperoleh dan modularitas jaringannya. Penelitian ini menggunakan metode louvain algoritma dalam mendeteksi komunitas yang terbentuk dari sebuah dataset.

Keywords— Betweenness Centrality, Community, Influential Users, Louvain Algorithm, Social Network Analysis, Twitter

I. INTRODUCTION

Media adalah salah satu bentuk kebutuhan yang selalu dikonsumsi oleh setiap individu. Media digunakan dengan berbagai macam tujuan dan motif yang berbeda-beda. Menurut McQuail (1987) motif individu menggunakan media memiliki empat indikator, diantaranya adalah informasi, identitas pribadi, integrasi dan interaksi sosial serta motif hiburan.

Kebutuhan dan kepuasan yang berbeda-beda dalam penggunaan media komunikasi tersebut menyebabkan komunikasi dipaksa berkembang lebih cepat. Akhirnya banyak media baru bermunculan dan berkembang diantara media-media lainnya. Salah satu jenis media baru yang kini sedang berkembang adalah internet. “Seperti yang diketahui internet (interconnection networking) merupakan bentuk konvergensi dari beberapa teknologi penting terdahulu, seperti komputer, televisi, radio dan telepon” (Bungin, 2006, p.135).

Begitu besarnya pengguna internet di dunia sehingga semakin banyak pula bermunculan situs-situs/ social media baru yang dapat diakses oleh pengguna internet. Salah satunya adalah Twitter sebagai jejaring sosial yang ada di dunia yang dioperasikan oleh lebih dari 328 juta pengguna aktif di seluruh dunia.

Para pengguna jejaring sosial Twitter ini dapat menulis status, mengunggah foto, sharing foto, dan sharing video. Dengan menggunakan twitter dapat dilihat community, circle dan influential user. Salah satu keunikan twitter ini adalah kemampuan untuk mendeteksi community yang dihasilkan dari pengguna twitter tanpa batas yang memiliki jumlah pertemanan tanpa memaksimalkan berapa jumlah banyak yang ada di dalam daftar pertemanan.

II. LITERATURE STUDY

A. Social Network

Social Network atau yang sering disebut Jejaring Sosial merupakan sebuah bentuk layanan internet atau situs yang ditujukan sebagai komunitas *online* bagi orang yang memiliki kesamaan aktivitas, ketertarikan pada bidang tertentu, berbagi informasi, dan berkomunikasi. Jejaring sosial terbesar antara lain adalah *Facebook*, *Twitter*, *Friendster*, *MySpace.com*, dan *Youtube*.

B. Information Network

Information Network atau Jaringan informasi merupakan suatu sistem terpadu yang bergerak dalam bidang pengolahan informasi dengan tujuan untuk menyediakan berbagai informasi dan sumber daya tanpa memperhatikan bentuk maupun asal data untuk keperluan masyarakat pemakai.

- Influential Users merupakan orang atau pengguna yang memiliki pengaruh besar di dalam jejaring sosial yang dapat membuat para masyarakat sekitar untuk melakukan sesuatu yang baik.
- Community Detection merupakan wadah yang nyaman untuk saling bertukar informasi antar komunitas baik informasi yang disebutkan secara eksplisit dan secara implisit.

C. Betweenness Centrality

Betweenness centrality yang akan menghitung bobot setiap node berdasar seberapa banyak node i dilalui oleh dua node lain dalam graf berdasar jalur terpendeknya. Perhitungan bobot *betweenness centrality* untuk setiap node dapat menggunakan rumus sebagai berikut :

$$C'_B(i) = \frac{2 \sum_{j < k} P_{jk}(i)}{(n-1)(n-2)} \quad [1]$$

$P_{jk}(i)$ adalah jumlah jalur terpendek antara node j dan k yang melewati i , sedangkan P_{jk} adalah jumlah jalur terpendek antara j dan k .

D. Breadth-First Search

Breadth-first search merupakan algoritma yang melakukan pencarian secara melebar yang mengunjungi suatu *node* kemudian mengunjungi semua *node* yang berhubungan dengan *node* tersebut terlebih dahulu. Dalam hal *influential users*, kita menggunakan algoritma *Breadth-First Search* (BFS) untuk graf tak berbobot.

E. Algorithms for Community Detection

Community Detection merupakan teknik yang digunakan untuk mendeteksi komunitas yang tercipta dari kumpulan dataset dengan melihat keterkaitan antara node (user) satu sama lain, biasanya kelompok simpul yang memiliki kemungkinan lebih tinggi untuk terhubung satu sama lain daripada dengan anggota kelompok lain, meskipun pola lain dimungkinkan. Tujuannya adalah untuk menemukan komunitas simpul yang erat yang memiliki ikatan persahabatan yang lebih jarang antara komunitas yang berbeda. Algoritma *community detection* pada penelitian ini menggunakan Louvain Algoritma. Rumus perhitungan dasar untuk mencari community menggunakan algoritma Louvain adalah menggunakan rumus (2).

$$Q = \frac{1}{2m} \sum_{i,j} \left[a_{i,j} - \frac{k_i k_j}{2m} \right] \delta(c_i c_j) \quad [2]$$

III. METHODS

A. Betweenness Centrality

Berdasarkan pengertian yang sudah dijelaskan sebelumnya, *Betweenness Centrality* dapat direpresentasikan dengan memperhatikan beberapa faktor yaitu :

- Jika ada dua aktor yang saling berdekatan, yaitu j dan k, ingin berinteraksi dan aktor i berada pada jalur hubungan antara j dan k, maka i memiliki kontrol terhadap interaksi keduanya.
- Betweenness mengukur kontrol tersebut.
- sehingga, jika i berada pada jalur dari beberapa interaksi, maka i adalah sebuah aktor penting.

B. Dataset Analysis

Penelitian ini menggunakan dataset twitter yang di crawled dari public resources oleh Stanford University (<http://snap.stanford.edu/data/egonets-Twitter.html>).

| Dataset Statistics | |
|--------------------------------|-----------------|
| Nodes | 81306 |
| Edges | 1342310 |
| Nodes in Largest WCC | 81306 (1.000) |
| Edges in largest WCC | 1768149 (1.000) |
| Nodes in largest SCC | 68413 (0.841) |
| Edges in largest SCC | 1685163 (0.953) |
| Average clustering coefficient | 0.5653 |
| Number of triangles | 13082506 |
| Fraction of closed triangles | 0.06415 |

```
[5] #Gambar Representasi Network
print (nx.info(G_Tw))
```

Gambar 1. Script info() dataset twitter

```
Name:
Type: Graph
Number of nodes: 4113
Number of edges: 56691
Average degree: 27.5667
```

Gambar 2. Hasil run code info()

Nodes merepresentasikan sebagai akun. Edges merepresentasikan sebagai hubungan *mem-follow* dua atau lebih akun.

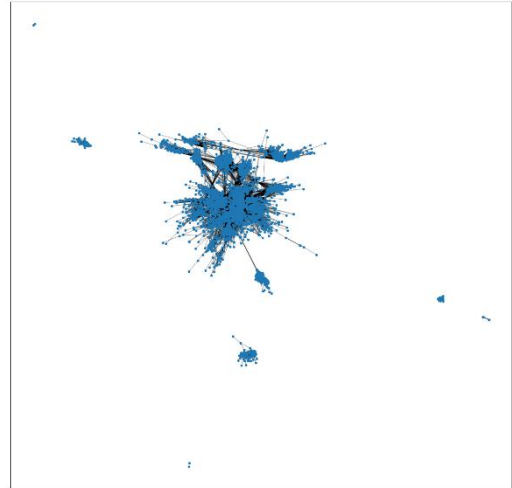
C. Connected Network

Dalam penelitian ini dari 81.306 node dan 1.342.310 edge dihasilkan graph yang merepresentasikan keterhubungan setiap nodes dapat dilihat pada gambar 4.

```
G_Tw = nx.read_edgelist
('/content/drive/My Drive/twitter_combined/twitter_combined.txt',
create_using = nx.Graph(), nodetype = int)

plt.figure(figsize=(20,20))
nx.draw_networkx(G_Tw)
plt.savefig('/content/drive/My Drive/graph.pdf')
```

Gambar 3. Script untuk merepresentasikan dataset menjadi sebuah graph.



Gambar 4. Hasil run code draw_networkx()

IV. RESULT

Pada penelitian dilakukan dengan memvisualisasikan dataset Twitter dengan menggunakan google colab. Data divisualisasikan ke dalam sociogram dimana titik pada gambar disebut 'node' atau simpul merepresentasikan seorang individu yang dihubungkan oleh garis yang disebut 'vertex'. Dua node yang terhubung dinyatakan dengan adanya garis yang menghubungkan keduanya. Untuk mengetahui node (individu) yang menjadi influential users, penelitian ini menggunakan algoritma betweenness. Hasil yang diperoleh dari algoritma tersebut dapat dilihat pada tabel 1.

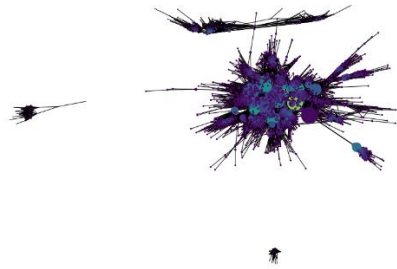
A. Influential Users

Dalam hal untuk menemukan *most influential users* di jejaring sosial ini dapat dilihat pada gambar 5.

```
pos = nx.spring_layout(G_fb)
betCent = nx.betweenness_centrality(G_fb, normalized=True, endpoints=True)
node_color = [20000.0 * G_fb.degree(v) for v in G_fb]
node_size = [v * 10000 for v in betCent.values()]
plt.figure(figsize=(20,20))
nx.draw_networkx(G_fb, pos=pos, with_labels=False,
node_color=node_color,
node_size=node_size )
plt.axis('off')
sorted(betCent, key=betCent.get, reverse=True)[:5]
```

Gambar 5. Script untuk merepresentasikan untuk menemukan most influential users dengan betweenness centrality.

Dan hasil run script diatas dapat dilihat pada gambar 6.



Gambar 6. Graph hasil 5 node teratas (most influential users) dengan betweenness Centrality

Pada tabel 1 dibawah dapat dilihat 5 node teratas yang menjadi *influential users* berdasarkan values terbesar.

Tabel 1. Hasil 5 node teratas yang menjadi *influential node*

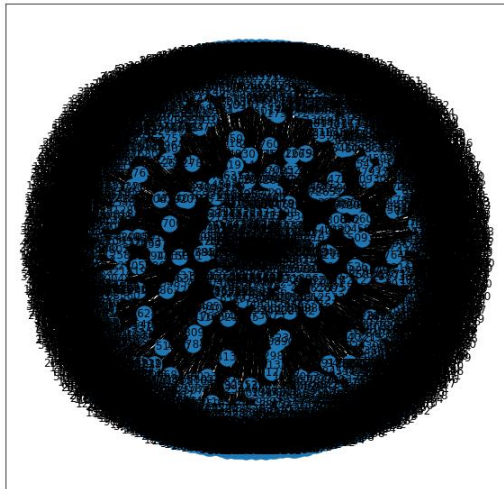
| No | Node | Values |
|----|----------|---------------------|
| 1. | 17093617 | 0.09213979977533762 |
| 2. | 783214 | 0.08562137017443493 |
| 3. | 41716273 | 0.08552761505870383 |
| 4. | 15666380 | 0.08013042325207921 |
| 5. | 3359851 | 0.07607026214734106 |

Untuk mendapatkan node teratas yang menjadi *influential users* berdasarkan values terbesar dapat dilihat pada gambar 7.

```
most_influential = nx.betweenness_centrality(G_fb)
for w in sorted(most_influential, key=most_influential.get, reverse=True)[:5]:
    print(w, most_influential[w])
```

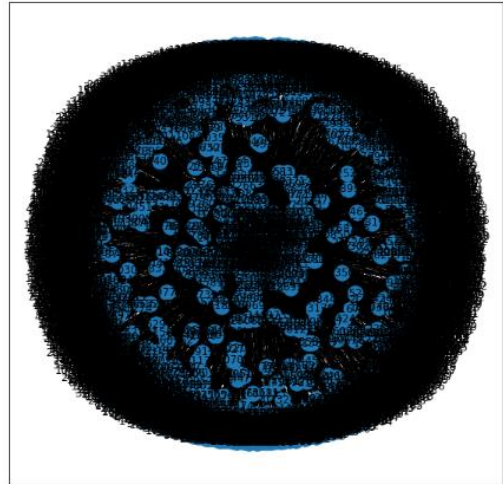
Gambar 7. Script untuk merepresentasikan untuk menemukan node teratas *influential users* berdasarkan values terbesar.

1. Hasil node “17093617” dengan menggunakan *breadth-first search* dengan values sebesar yaitu 0.09213979977533762 dapat dilihat pada gambar 8.



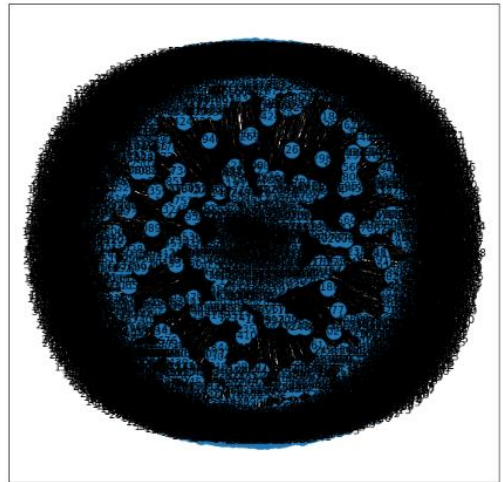
Gambar 8. Connection Node “17093617”

2. Hasil node “783214” dengan menggunakan *breadth-first search* dengan values sebesar yaitu 0.08562137017443493 dapat dilihat pada gambar 9.



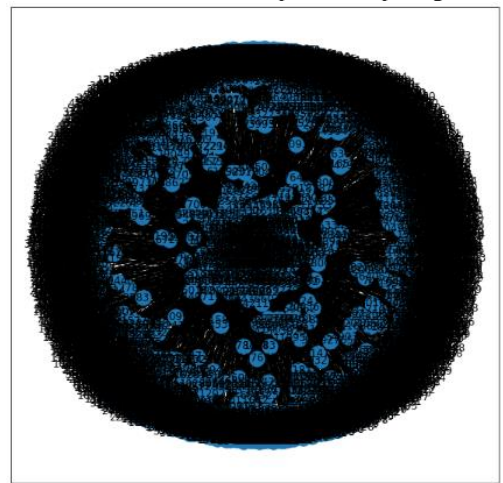
Gambar 9. Connection Node “783214”

3. Hasil node “41716273” dengan menggunakan *breadth-first search* dengan values sebesar yaitu 0.08552761505870383 dapat dilihat pada gambar 10.



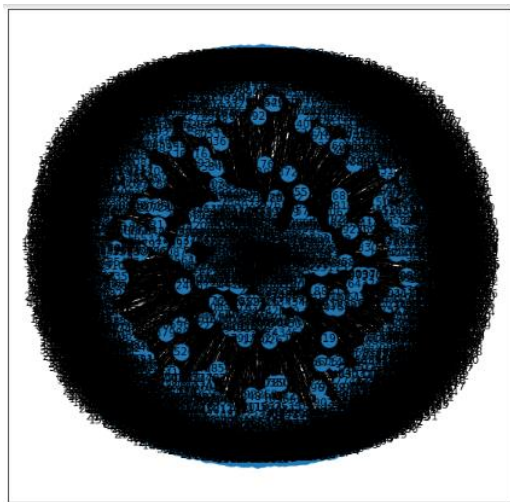
Gambar 10. Connection Node “41716273”

4. Hasil node “15666380” dengan menggunakan *breadth-first search* dengan values sebesar yaitu 0.08013042325207921 dapat dilihat pada gambar 11.



Gambar 11. Connection Node “15666380”

5. Hasil node “3359851” dengan menggunakan *breadth-first search* dengan values terbesar yaitu 0.07607026214734106 dapat dilihat pada gambar 12.



Gambar 12. Connection Node “3359851”

```
G_comm.add_nodes_from(dict_nodes)

# Calculating modularity and the total number of communities
mod=community.modularity(partition,G_Tw)
print("Modularity: ", mod)
print("Total number of Communities=", len(G_comm.nodes()))

# Creating the Graph and also calculating Modularity
matplotlib.rcParams['figure.figsize'] = [12, 8]
pos_louvain=nx.spring_layout(G_comm)
nx.draw_networkx(G_comm, pos_louvain, with_labels=True, node_size=100, font_size=11, label='Modularity = ' + str(round(mod,3))
', Communities= ' + str(len(G_comm.nodes())))

plt.suptitle('Community structure (Louvain Algorithm)', fontsize=22, fontname='Arial')
plt.box(on=None)
plt.axis('off')
plt.legend(bbox_to_anchor=(0,1), loc='best', ncol=1)
plt.savefig('louvain.png', dpi=400, bbox_inches='tight')
```

Modularity: 0.8568647399281242
Total number of Communities= 24

Gambar 15. Script untuk menampilkan community yang tercipta dari keseluruhan dataset.



Gambar 16. Hasil akhir community yang dihasilkan.

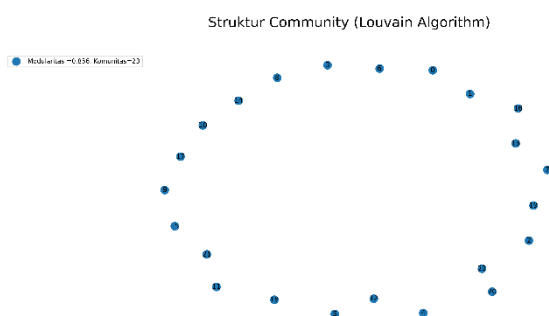
B. Community Detection

```
partition=community.best_partition(G_Tw, weight='MsgCount')
print('Completed Louvain algorithm . . . ')
values=[partition.get(node) for node in G_Tw.nodes()]
list_com=partition.values()

# Creating a dictionary like {community_number:list_of_participants}
dict_nodes={}

# Populating the dictionary with items
for each_item in partition.items():
    community_num=each_item[1]
    community_node=each_item[0]
    if community_num in dict_nodes:
        value=str(dict_nodes.get(community_num)) + ' | ' + str(community_node)
        dict_nodes.update({community_num:value})
    else:
        dict_nodes.update({community_num:community_node})
```

Gambar 13. Script Louvain Algorithm untuk menampilkan struktur community



Gambar 14. Hasil dari algoritma louvain

V. KESIMPULAN

- Perhitungan centrality social network Twitter dilihat dari segi friends, yaitu di-follow dan mem-follow telah mampu mewakili proses perhitungan dalam penentuan central dari suatu jaringan.
- Metode betweenness centrality digunakan ketika user ingin mengetahui node atau akun Twitter mana yang memiliki pengaruh paling kuat di dalam suatu jaringan, sehingga akun Twitter tersebut mampu membuat koneksi ke pasangan atau kelompok lain dalam suatu jaringan.
- Terdapat 5 node teratas yang merepresentasikan influentials user paling berpengaruh.
- Pada penelitian ini deteksi komunitas ditentukan menggunakan algoritma louvain dan menghasilkan 23 komunitas dari dataset yang ada.

REFERENCES

- [1] Bader, D.A., Kintali, S., Madduri, K. & Mihail, M. (2007). Approximating Betweenness Centrality. Springer-Verlag, 124-137.
- [2] Hanneman, R.A. & Riddle, M. (2005). Introduction to Social Network Methods. Riverside, CA: University of California, Riverside, Online textbook.
- [3] Tang, L. & Liu, H. (2010). Community Detection and Mining in Social Media. United States : Morgan & Claypool Publishers.
- [4] P. De Meo, E. Ferrara, G. Fiumara, A. Provetti. Generalized Louvain Method for Community Detection in Large Scale. 11th International Conference on Intelligent System Design and Application (2011) 88-93