

Perbandingan Algoritme Dijkstra dan *Node Combination* Dalam Perhitungan *Betweenness Centrality* pada Graf Jaringan Listrik Universitas Padjadjaran Jatinangor

JEANE R.M. D.P. CHANTIQUE, HERLINA NAPITUPULU, BETTY SUBARTINI

Program Studi S-1 Matematika, Fakultas MIPA, Universitas Padjadjaran
Jl. Raya Bandung Sumedang KM 21 Jatinangor Sumedang 45363
jeane17001@mail.unpad.ac.id, herlina@unpad.ac.id,
betty.subartini@unpad.ac.id

Abstrak

Listrik sangat penting untuk memenuhi kebutuhan masyarakat secara umum dan terkhusus instansi pendidikan yang kini mengandalkan teknologi dalam proses pembelajaran. Terjadinya pemadaman listrik tentunya menghambat proses pembelajaran di suatu instansi pendidikan. Salah satu faktor penyebab terjadinya pemadaman listrik karena adanya kerusakan jaringan distribusi listrik. Menganalisis kekritisan gardu sebagai bagian dari jaringan distribusi listrik dapat dilakukan dengan cara menghitung nilai sentralitas (*centrality*) dari suatu gardu. Pada penelitian ini dicari gardu yang berpengaruh paling besar dalam jaringan listrik di Universitas Padjadjaran Jatinangor. Adapun perhitungan *centrality* yang digunakan adalah *betweenness centrality* yang melibatkan perhitungan lintasan terpendek di dalamnya. Dua buah algoritma pencarian lintasan terpendek digunakan, yakni algoritme Dijkstra dan algoritme *Node Combination*, untuk dibandingkan performa maupun hasilnya untuk kasus graf berarah yang diteliti yakni jaringan listrik Unpad Jatinangor. Perhitungan *betweenness centrality* juga dilakukan dengan bantuan *software* Python dan diperoleh gardu yang paling berpengaruh adalah gardu listrik di Fakultas Matematika dan Ilmu Pengetahuan Alam dan gardu listrik di Fakultas Peternakan dengan nilai *betweenness centrality* sebesar 0.09091.

Kata Kunci: Listrik, Pemadaman Listrik, *Centrality*, Algoritme Dijkstra, Algoritme *Node Combination*

Abstract

Electricity is very important for the needs of society in general and especially educational institutions that now rely on technology in the learning process. The occurrence of a power outage certainly hampers the learning process in an educational institution. One of the factors causing power outages is due to damage to the electricity distribution network. Analyzing the criticality of the substation as part of the electricity distribution network can be done by calculating the centrality value of a substation. In this research, the substation that has the greatest influence on the electricity network is sought at Padjadjaran University, Jatinangor. The centrality calculation used is betweenness centrality which involves calculating the shortest path in it. Two shortest path search algorithms are used, namely Dijkstra's algorithm and Node Combination algorithm, to compare performance and results for the directed graph case studied, namely the Jatinangor Unpad electricity network. The calculation of betweenness centrality was also carried out with the help of Python software and the most influential substations were electrical substations at the Faculty of Mathematics and Natural Sciences and electrical substations at the Faculty of Animal Science with a betweenness centrality value of 0.09091.

Keywords: Electricity, Power Outage, Centrality, Dijkstras algorithm, Node Combination algorithm

1. PENDAHULUAN

Saat ini, peranan listrik dalam kelangsungan hidup manusia sangat dominan. Dari mulai lampu penerangan, proses produksi yang melibatkan mesin-mesin industri hingga sarana pendidikan semuanya mengandalkan listrik. Instansi pendidikan kini memanfaatkan teknologi dalam proses pembelajaran. Terjadinya gangguan listrik seperti adanya pemadaman listrik tentu menghambat proses pembelajaran dalam suatu instansi pendidikan. Rusaknya gardu pemasok listrik pada satu wilayah pun menjadi salah satu dari sekian banyak faktor yang mungkin menjadi penyebab adanya pemadaman listrik. Oleh karena itu, penulis ingin menganalisis kekritisitas simpul yang merepresentasikan gardu yang saling terhubung yang bertujuan untuk menjamin kekuatan jaringan listrik tanpa mengurangi kinerjanya dengan menghitung nilai *betweenness centrality* dari tiap gardu.

Betweenness centrality melihat keantaraan antar simpul, seberapa sering sebuah simpul dilewati oleh simpul lainnya. Untuk menghitung *betweenness centrality* dibutuhkan pencarian lintasan terpendek terlebih dahulu. Pada penelitian ini, digunakan dua cara dalam pencarian lintasan terpendek, yaitu algoritme Dijkstra dan algoritme *Node Combination*. Pada penelitian ini diberikan hasil analisis *betweenness centrality* dalam meninjau kekritisitas tiap-tiap gardu dalam jaringan listrik Unpad Jatinangor. Penelitian serupa pernah dilakukan oleh Nguyen [8] yang menganalisis kekritisitas simpul pada jaringan listrik yang saling bergantung menggunakan tiga metode yaitu metode *Maximum Cascading* (Max-Cas), metode *Iterative Interdependent Centrality* (IIC), dan metode *Hybrid*. Selain itu, penelitian serupa pernah dilakukan oleh Roy *et al* [10] dengan menganalisis kesentralan pada jaringan transportasi di luar struktur menggunakan *betweenness centrality*, *closeness centrality*, dan *degree centrality*. Di Indonesia, Yunita *et al* [12] mengkaji sosial network twitter untuk menentukan pengguna yang berpengaruh dalam penyebaran informasi menggunakan *betweenness centrality*. Selain itu, Susanto *et al* [12] yang menganalisis *centrality* menggunakan *betweenness* dan *closeness centrality* pada *social network* twitter.

Sebelum masuk ke pembahasan lebih lanjut akan dikenalkan terlebih dahulu beberapa materi yang digunakan dalam penelitian ini.

1.1. Graf dan Lintasan Terpendek. Suatu graf G didefinisikan sebagai pasangan himpunan (V, E) yang ditulis dengan notasi $G = (V, E)$, dimana V adalah himpunan tidak kosong dari simpul-simpul dan E adalah himpunan sisi yang menghubungkan sepasang simpul (Munir [7]).

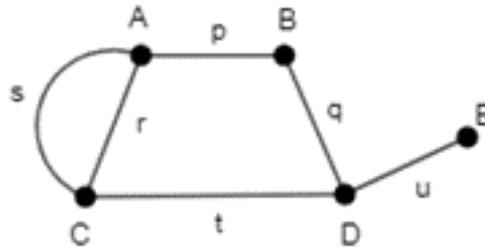


FIGURE 1. Contoh graf

Pencarian lintasan terpendek untuk menemukan jalur antara dua simpul dalam suatu graf sehingga jumlah bobotnya minimum. Bobot pada graf bisa berupa jarak, waktu, biaya, dan sebagainya. Lubis [4] menyatakan bahwa biasanya bobot yang ada pada graf berupa nilai positif, namun tidak menutup kemungkinan terdapat nilai yang negatif. Jika graf yang digunakan bukan graf berbobot, maka nilai bobot sisi dari satu simpul ke simpul lainnya adalah 1 (Madkour *et al*, [6]).

1.2. Algoritme Dijkstra. Harahap [2] menyatakan bahwa algoritme dijkstra merupakan algoritme untuk menentukan jarak terpendek dari satu simpul ke simpul lainnya dalam suatu graf berbobot, dimana jarak antar simpul merupakan bobot dari setiap sisi pada graf dan bobot tidak boleh bernilai negatif. Menurut Maw [7], langkah-langkah algoritme dijkstra yaitu:

Langkah 1. Inisiasi

- Pilih simpul s sebagai simpul awal.
- Simpul awal (simpul s) adalah $(0, p)$ yang diberi label permanen dan bobotnya adalah $0(d(s) = 0)$.
- Setiap simpul lain (∞, t) diberi label sementara dan bobotnya ∞ .

Langkah 2. Menentukan jarak (bobot) dari simpul

- Misal $d(i)$ merupakan bobot dari simpul awal. Cari simpul dengan label sementara (∞, t) dari simpul (d_i) dengan simpul yang bertetangga (i, j) . Tentukan bobot dari simpul tersebut.
- $d_j = \min \{d_j, d_i + d_{ij}\}$ dimana d_{ij} adalah bobot antara simpul ke i dan ke j .
- Pilih simpul dengan bobot terkecil dan beri label permanen.

Langkah 3. Terminasi

- Jika tidak terdapat simpul yang dapat dijangkau dari simpul s , maka langkah selesai.
- Jika tidak dapat menjangkau simpul berlabel sementara dari simpul saat ini, maka semua label sementara jadi permanen.
- Jika tidak, kembali ke langkah 2.

1.3. Algoritme *Node Combination*. Algoritme *Node Combination* diperkenalkan oleh Lu dan Camidz pada tahun 2011 dengan ide dasar dari algoritme ini yaitu menggabungkan simpul sumber dengan simpul tetangganya secara iteratif menjadi simpul sumber baru dan memperbarui bobot sisi dari simpul yang tersisa. Langkah-langkah algoritme *Node Combination* menurut Lu [5] adalah sebagai berikut.

Langkah 1.: Inisialisasikan $d(v_s) = 0$ dengan v_s merupakan simpul awal atau simpul sumber.

Langkah 2.: Temukan node tetangga terdekatnya. Pilih v_k dari tetangga v_s , yang membuat $w_{sk} = \min \{w_{sj}\}$, misalkan $d(v_k) = w_{sk}$. Jika tidak ada simpul yang bertetangga terhadap v_s , berhenti.

Langkah 3.: Menggabungkan node. Hapus v_k , $V = V - v_k$. Jika $V = \emptyset$, berhenti.

Langkah 4.: Memodifikasi bobot sisi. Untuk setiap sisi e_{kj} , perbarui $w_{sj} = \min\{w_{sj}, w_{sk} + w_{kj}\}$. Lanjutkan ke langkah 1.

1.4. **Betweenness Centrality.** *Betweenness centrality* melihat keantaraan antar simpul dan tidak mempertimbangkan bobot simpul dalam perhitungannya. *Betweenness centrality* untuk setiap simpul adalah banyaknya jalur terpendek yang melewati simpul tersebut (Freeman, [1]). *Betweenness centrality* bergantung pada banyaknya simpul yang ada pada jaringan. Pada jaringan listrik, *betweenness centrality* dimanfaatkan untuk mengidentifikasi kekritisian elemen jaringan listrik, dimana elemen yang nilai *betweenness centrality* paling tinggi artinya elemen tersebut lebih rentan dan berpengaruh terhadap elemen jaringan listrik yang lain apabila terjadi gangguan dalam suatu jaringan listrik tersebut (Kong *et al*, [3]). Adapun rumus untuk menghitung *betweenness centrality* untuk graf berarah yaitu:

$$C'_B(v) = \frac{1}{(n-1)(n-2)} \sum_{v \neq s \neq t \in V} \frac{\sigma_{s,t}(v)}{\sigma_{s,t}} \quad (1)$$

dimana

$\sigma_{s,t}$: banyaknya lintasan terpendek dari simpul s ke t

$\sigma_{s,t}(v)$: banyaknya lintasan terpendek dari simpul s ke t yang melewati simpul v

n : banyaknya simpul pada graf

2. METODE PENELITIAN

Metode yang digunakan dalam pencarian lintasan terpendek pada jaringan listrik Universitas Padjadjaran Jatinangor dengan menggunakan algoritme Dijkstra dan *Node Combination*, sedangkan metode yang digunakan untuk menganalisis kekritisian gardu pada jaringan listrik Universitas Padjadjaran Jatinangor yaitu dengan menghitung *centrality* dari tiap gardu dimana *centrality* yang digunakan yaitu *betweenness centrality*. Pengolahan data dilakukan dengan bantuan *software* Python.

3. HASIL DAN PEMBAHASAN

Pada bab ini dibahas perbandingan algoritme Dijkstra dan algoritme *Node Combination* dalam pencarian lintasan terpendek pada graf jaringan listrik Unpad Jatinangor, hasil perhitungan *betweenness centrality* dari data yang sudah diperoleh, implementasi *betweenness centrality* pada graf jaringan listrik Unpad Jatinangor, dan visualisasi hasil perhitungan *betweenness centrality*.

Pada Gambar 2 ditampilkan peta diagram jaringan listrik Unpad Jatinangor. Kemudian dari peta diagram tersebut direpresentasikan ke dalam bentuk graf berarah yang ditampilkan pada Gambar 3. Pada penelitian ini, simpul merepresentasikan gardu, sedangkan sisi merepresentasikan aliran listrik antar gardu. Bobot yang digunakan bernilai 1 jika adanya aliran listrik, 0 jika tidak.

3.1. Pencarian Lintasan Terpendek Menggunakan Algoritma Dijkstra. Pada Tabel 1 ditampilkan pencarian lintasan terpendek dari graf jaringan listrik Kampus Universitas Padjadjaran Jatinangor dengan menggunakan Algoritme Dijkstra.

Keterangan:



FIGURE 2. Peta diagram jaringan listrik Unpad Jatininggor

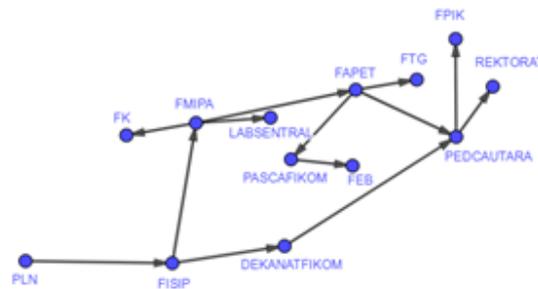


FIGURE 3. Graf jaringan listrik Unpad Jatininggor

PLN= Gardu Perusahaan Listrik Negara

A= Gardu Fakultas Ilmu Sosial dan Ilmu Politik

B= Gardu Fakultas Matematika dan Ilmu Pengetahuan Alam

C= Gardu Fakultas Kedokteran

D= Gardu Laboratorium Sentral

E= Gardu Fakultas Peternakan

F= Gardu Pasca Fakultas Ilmu Komunikasi

G= Gardu Fakultas Teknik Geologi

H= Gardu Fakultas Ekonomi dan Bisnis

I= Gardu Fakultas Pedca Utara

J= Gardu Fakultas Perikanan dan Ilmu Kelautan

K= Gardu Gedung Rektorat

L= Gardu Gedung Dekanat Fakultas Ilmu Komunikasi

TABLE 1. Pencarian lintasan terpendek menggunakan menggunakan algoritme Dijkstra

Iterasi	Simpul yang dipilih	Bobot												
		PLN	A	B	C	D	E	F	G	H	I	J	K	L
1	PLN	0	1	∞										
2	A	0	1	2	∞	2								
3	B	0	1	2	3	3	3	∞	∞	∞	∞	∞	∞	2
4	L	0	1	2	3	3	3	∞	∞	∞	3	∞	∞	2
5	C	0	1	2	3	3	3	∞	∞	∞	3	∞	∞	2
6	D	0	1	2	3	3	3	∞	∞	∞	3	∞	∞	2
7	E	0	1	2	3	3	3	4	4	∞	3	∞	∞	2
8	I	0	1	2	3	3	3	4	4	∞	3	4	4	2
9	F	0	1	2	3	3	3	4	4	5	3	4	4	2
10	G	0	1	2	3	3	3	4	4	5	3	4	4	2
11	J	0	1	2	3	3	3	4	4	5	3	4	4	2
12	K	0	1	2	3	3	3	4	4	5	3	4	4	2
13	H	0	1	2	3	3	3	4	4	5	3	4	4	2

Diperoleh lintasan terpendek dari:

- Gardu PLN ke Gardu FISIP yaitu PLN-FISIP dengan bobot 1.
- Gardu PLN ke Gardu FMIPA yaitu PLN-FISIP-FMIPA dengan bobot 2.
- Gardu PLN ke Gardu FK yaitu PLN-FISIP-FMIPA-FK dengan bobot 3.
- Gardu PLN ke Gardu LabSentral yaitu PLN-FISIP-FMIPA-LabSentral dengan bobot 3.
- Gardu PLN ke Gardu FAPET yaitu PLN-FISIP-FMIPA-FAPET dengan bobot 3.
- Gardu PLN ke Gardu Pasca FIKOM yaitu PLN-FISIP-FMIPA-FAPET-Pasca FIKOM dengan bobot 4.
- Gardu PLN ke Gardu FTG yaitu PLN-FISIP-FMIPA-FAPET-FTG dengan bobot 4.
- Gardu PLN ke Gardu FEB yaitu PLN-FISIP-FMIPA-FAPET-Pasca FIKOM-FEB dengan bobot 5.
- Gardu PLN ke Gardu Pedca Utara yaitu PLN-FISIP-Dekanat FIKOM-Pedca Utara dengan bobot 3.
- Gardu PLN ke Gardu FEB yaitu PLN-FISIP-FMIPA-FAPET-Pasca FIKOM-FEB dengan bobot 5.
- Gardu PLN ke Gardu FPIK yaitu PLN-FISIP-Dekanat FIKOM-Pedca Utara-FPIK dengan bobot 4.
- Gardu PLN ke Gardu Rektorat yaitu PLN-FISIP-Dekanat FIKOM-Pedca Utara-Rektorat dengan bobot 4.
- Gardu PLN ke Gardu Dekanat FIKOM yaitu PLN-FISIP-Dekanat FIKOM dengan bobot 2.

3.2. Pencarian Lintasan Terpendek Menggunakan Algoritme *Node Combination*.

Pencarian lintasan terpendek dari gardu PLN ke gardu lainnya menggunakan Algoritme *Node Combination* untuk graf jaringan listrik Unpad Jatiningor ditampilkan pada Tabel 2.

TABLE 2. Pencarian lintasan terpendek menggunakan menggunakan algoritme Dijkstra

Iterasi	v_s	v_k	Bobot												
			PLN	A	B	C	D	E	F	G	H	I	J	K	L
1	PLN	-	0	1	∞										
2	PLN	A	0	1	2	∞	2								
3	PLN	B	0	1	2	3	3	3	∞	∞	∞	∞	∞	∞	2
4	PLN	L	0	1	2	3	3	3	∞	∞	∞	3	∞	∞	2
5	PLN	E	0	1	2	3	3	3	4	4	∞	3	∞	∞	2
6	PLN	I	0	1	2	3	3	3	4	4	∞	3	4	4	2
7	PLN	F	0	1	2	3	3	3	4	4	5	3	4	4	2

3.3. Perhitungan *Betweenness Centrality* pada Data Jaringan Listrik Unpad Jatinangor. *Betweenness centrality* menghitung seberapa sering sebuah simpul dilewati oleh simpul lainnya di dalam suatu jaringan. Simpul yang memiliki nilai *betweenness* yang tinggi atau besar memiliki pengaruh yang kuat dalam suatu jaringan. Sebaliknya, simpul yang memiliki nilai *betweenness* yang rendah atau kecil memiliki pengaruh yang sedikit dalam suatu jaringan. Dalam jaringan listrik, *betweenness centrality* digunakan untuk mengidentifikasi kekritisan gardu pada jaringan listrik. Pada bagian ini diberikan perhitungan *betweenness centrality* secara manual dan dengan bantuan *software* Python.

TABLE 3. Perhitungan *Betweenness Centrality* pada gardu B (FMIPA)

Jalur (s, t)	Lintasan terpendek	$\sigma_{s,t}$	$\sigma_{s,t}(B)$	$\frac{\sigma_{s,t}(B)}{\sigma_{s,t}}$
(PLN,A)	PLN-A	1	0	0
(PLN,C)	PLN-A-B-C	1	1	1
(PLN,D)	PLN-A-B-D	1	1	1
(PLN,E)	PLN-A-B-E	1	1	1
(PLN,F)	PLN-A-B-E-F	1	1	1
(PLN,G)	PLN-A-B-E-G	1	1	1
(PLN,H)	PLN-A-B-E-F-H	1	1	1
(PLN,I)	PLN-A-L-I	1	0	0
(PLN,J)	PLN-A-L-I-J	1	0	0
(PLN,K)	PLN-A-L-I-K	1	0	0
(PLN,L)	PLN-A-L	1	0	0
(A,C)	A-B-C	1	1	1
(A,D)	A-B-D	1	1	1
(A,E)	A-B-E	1	1	1
(A,F)	A-B-E-F	1	1	1
(A,G)	A-B-E-G	1	1	1
(E,H)	E-F-H	1	0	0
(E,I)	E-I	1	0	0
(E,J)	E-I-J	1	0	0
(E,K)	E-I-K	1	0	0
(F,H)	F-H	1	0	0
(I,J)	I-J	1	0	0
(I,K)	I-K	1	0	0
(L,I)	L-I	1	0	0
(L,J)	L-I-J	1	0	0
(L,K)	L-I-K	1	0	0
$\sum_{F \neq s \neq t \in V} \frac{\sigma_{s,t}(B)}{\sigma_{s,t}}$				12

Pada Tabel 3 ditampilkan contoh perhitungan nilai *betweenness centrality* secara manual dari salah satu gardu yaitu gardu B atau gardu listrik di Fakultas Matematika dan Ilmu Pengetahuan Alam.

$$C'_B = \frac{1}{(n-1)(n-2)} \sum_{s \neq t \neq v \in V} \frac{\sigma_{st}(B)}{\sigma_{st}}$$

$$C'_B = \frac{1}{(13-1)(13-2)}(12) = 0.090909 \approx 0.09091$$

Sehingga, nilai *betweenness centrality* pada gardu FMIPA yaitu 0.09091.

Pada Tabel 4 ditampilkan perhitungan nilai *betweenness centrality* menggunakan *software* Python. Nilai *betweenness centrality* yang bercetak tebal pada Tabel 4 menunjukkan nilai *betweenness centrality* yang terbesar.

TABLE 4. Perhitungan *Betweenness Centrality* pada Gardu B (FMIPA) Menggunakan *Python*

Gardu	Nilai <i>Betweenness Centrality</i>
PLN	0
FISIP	0.08333
FMIPA	0.09091
FK	0
Lab. Sentral	0
FAPET	0.09091
Pasca FIKOM	0.03030
FTG	0
FEB	0
Pedca utara	0.075756
FPIK	0
Rektorat	0
Dekanat	0
FIKOM	0

Berdasarkan Tabel 4, diperoleh nilai *betweenness centrality* terbesar ada pada gardu listrik di FMIPA dan FAPET yang bernilai 0.09091. Sedangkan nilai *betweenness centrality* terkecil ada pada gardu PLN, FK, Lab.Sentral, FTG, FEB, FPIK, dan Rektorat dengan nilai *betweenness centrality* bernilai sama yaitu 0. Artinya, gardu yang berada di FMIPA dan FAPET menjadi gardu yang berperan penting dalam jaringan listrik Unpad Jatinangor, dikarenakan kedua gardu tersebut memiliki nilai *betweenness centrality* yang paling tinggi dibandingkan dengan gardu yang lainnya.

4. SIMPULAN

Nilai *betweenness centrality* tertinggi ada pada gardu listrik di Fakultas Matematika dan Ilmu Pengetahuan Alam dan gardu listrik di Fakultas Peternakan dengan nilai *betweenness centrality* sebesar 0.09091. Sedangkan nilai *betweenness centrality* terkecil ada pada gardu PLN, gardu Fakultas Kedokteran, gardu Laboratorium Sentral, gardu Fakultas Teknik Geologi, gardu Fakultas Ekonomi dan Bisnis, gardu Fakultas Perikanan dan Ilmu Kelautan, gardu Gedung Rektorat, dengan nilai *betweenness centrality* bernilai sama yaitu 0. Artinya, gardu listrik di Fakultas Matematika dan Ilmu Pengetahuan Alam dan gardu listrik di Fakultas Peternakan menjadi gardu yang berpengaruh dan berperan penting dalam jaringan listrik Universitas Padjadjaran Jatinangor, dikarenakan gardu tersebut memiliki nilai *betweenness centrality* yang paling tinggi dibandingkan dengan gardu yang lainnya. Sehingga, pihak Universitas Padjadjaran dapat lebih memperhatikan pemeliharaan gardu FMIPA dan FAPET dalam upaya mitigasi pemadaman listrik di Universitas Padjadjaran Jatinangor.

Ucapan Terimakasih.

Terimakasih sebesar-besarnya kepada sarana prasarana Universitas Padjadjaran Jatinangor yang telah memberikan izin dilaksanakannya penelitian dengan Nomor Surat 4044/UN6.D.1/PK.01.06/2021.

REFERENCES

- [1] Freeman, L.C., 1978, Centrality in Social Networks Conceptual Clarification, *Social Networks*, Vol.1 (3):215-239.
- [2] Harahap, M. K. dan Khairina, N., 2017, Pencarian Jalur Terpendek dengan Algoritma Dijkstra, *Sinkron : Jurnal Dan Penelitian Teknik Informatika*, 2(2), 18-23.
- [3] Kong, Han, Guo and Pei. 2013, *An Energy-Based Centrality for Electrical Network*, Energy and Power Engineering, Vol.5:597-602
- [4] Lubis, Henny Syahriza. 2009, *Perbandingan Algoritma Greedy dan Dijkstra Untuk Menentukan Lintasan Terpendek*, Skripsi. Universitas Sumatera Utara.
- [5] Lu and Camitz. 2011, Finding the Shortest Paths by Node Combination, *Applied Mathematics and Computation* 217: 6401-6408.
- [6] Madkour, A., Aref, W.G., Rehman, F., Rahman, M.A., and Basalamah, S., 2017, *A survey of shortest-path algorithms*, arXiv preprint arXiv:1705.02044.
- [7] Munir, R. 2010, *Matematika Diskrit*. Informatika. Bandung.
- [8] Nguyen and Shen, 2013, Detecting Critical Nodes in Interdependent Power Networks for Vulnerability Assessment, *IEEE Transactions on Smart Grid*, Vol. 4 (1): 151-158.
- [9] Roy, Lim, and Zhu. 2015, Measuring centralities for transportation networks beyond structures. *Applications of Social Media and Social Network Analysis*, 23-39.
- [10] San san Maw, K.S.L. and Naing, L.L., 2019, IJMTT Call for Paper June-2020 UGC Approved Journal in 2017.
- [11] Susanto, Lina, dan Rachmat, 2012, Penerapan Social Network Analysis dalam Penentuan Centrality Studi Kasus Social Network Twitter, *Jurnal Informatika*, Vol. 8(1).
- [12] Yunita, Maharani, dan Gozali, 2015, Analisis dan Implementasi Betweenness Centrality pada Social Network Twitter dengan Metode Linear Scaling Berbasis Pengguna. *E-proceeding of Engineering*, Vol. 2 (1):1435-1445.

