

Visualisasi Pemain Sepak Bola Indonesia pada DBpedia dengan menggunakan Node2Vec dan Closeness Centrality

Ahmad Naufal Rofiif¹, Ardha Perwiradewa², Nur Aini Rakhmawati³

^{1,2,3}Departemen Sistem Informasi, Fakultas Teknologi Elektro dan Informatika Cerdas,
Institut Teknologi Sepuluh Nopember Surabaya

Jl. Raya ITS, Sukolilo, Surabaya 60111, Jawa Timur, Indonesia

Email: ¹rofiif16@mhs.is.its.ac.id, ²ardha.perwiradewa16@mhs.is.its.ac.id, ³nur.aini@is.its.ac.id

Abstract. *Visualization of Indonesian Football Players on DBpedia through Node2Vec and Closeness Centrality Implementation.* Through Semantic Web, data available on the internet are connected in a large graph. Those data are still raw so that they need to be processed to be an information that can help humans. This research aims to process and analyze the Indonesian soccer player graph by implementing node2vec and closeness centrality algorithm. The graph is modeled through a dataset obtained from the DBpedia by performing a SPARQL query on the SPARQL endpoint. The results of the Node2vec algorithm and closeness centrality are visualized for further analysis. Visualization of node2vec shows that the defenders are distributed over the players. Meanwhile, the result of closeness centrality shows that the strikers have the highest centrality score compared to other positions.

Keywords: *visualization, node2vec, closeness centrality*

Abstrak. Dengan adanya web semantik, data yang tersebar di internet dapat saling terhubung dan membentuk suatu graf. Data yang ada pada graf tersebut masih berupa data mentah sehingga perlu dilakukan pengolahan agar data mentah tersebut dapat menjadi informasi yang dapat membantu manusia. Penelitian ini bertujuan untuk melakukan pengolahan dan analisis terhadap graf pemain sepak bola Indonesia dengan mengimplementasikan algoritma node2vec dan closeness centrality. Graf dimodelkan melalui dataset yang didapat dari website DBpedia dengan cara melakukan query SPARQL pada SPARQL endpoint. Hasil dari algoritma node2vec dan closeness centrality divisualisasikan untuk dianalisis. Visualisasi dari node2vec menunjukkan pemain defender tersebar. Hasil closeness centrality menunjukkan bahwa pemain striker memiliki nilai tertinggi daripada posisi lainnya.

Kata Kunci: *visualisasi, node2vec, closeness centrality*

1. Pendahuluan

Saat ini berbagai macam data dan informasi bisa didapatkan dengan mudah melalui internet. *Semantic web* adalah sebuah konsep *website* yang menerapkan *linked data* dalam menyediakan informasi dan menggunakan *Resource Description Framework* (RDF) sebagai model data inti dalam *Linked Data* [1][2]. Dengan menerapkan *semantic web* maka data dan informasi yang tersebar di internet dapat saling terhubung dan membentuk suatu jaringan/ graf dengan berbagai macam data dan informasi didalamnya. Data yang ada pada graf tersebut masih berupa data mentah (*raw data*) sehingga perlu dilakukan pengolahan dan analisis agar data mentah tersebut bisa menjadi suatu *knowledge* yang dapat membantu manusia. Salah satu contoh *website* yang telah menerapkan *semantic web* adalah DBpedia. Penelitian ini melakukan analisa data "Pemain Sepak Bola Indonesia" yang ada di *website* <http://id.dbpedia.org/>. Analisa ini dilakukan untuk melihat kedekatan antar pemain dilihat dari sisi posisi dan klub.

Sentralitas menentukan seberapa pentingnya suatu *node* hanya berdasarkan struktur graf [3]. *Closeness centrality* adalah salah satu algoritma yang dapat digunakan untuk mengukur sentralitas pada suatu graf. *Closeness centrality* menghitung nilai sentralitas berdasarkan seberapa dekat suatu *node* dengan *node* lainnya [4]. Algoritma *closeness centrality* dipilih karena algoritma ini dapat menunjukkan *node* yang menjadi pusat karena kedekatannya dengan *node-node* lainnya. Pada penelitian sebelumnya sudah dilakukan eksplorasi terhadap *library* NetworkX mengenai bagaimana cara membuat graf dan melakukan implementasi beberapa

algoritma untuk melihat karakteristik dan sifat dari graf [5]. Algoritma ini telah digunakan untuk melihat hubungan aktor [6], [7], partai politik[8] serta siswa SMU di Instagram [9].

Penelitian ini membahas tentang hubungan antar *node* yang ada pada *dataset* pemain sepak bola Indonesia. Dengan menggunakan algoritma Node2Vec dan *closeness centrality* peneliti berusaha memetakan hubungan antar *node*: *node* Pemain, Klub dan Posisi. Data yang digunakan peneliti didapatkan melalui pembuatan *query* menggunakan SPARQL[5]. Selanjutnya, dilakukan pengukuran data menggunakan algoritma Node2Vec dan *closeness centrality*. Penggunaan algoritma *closeness centrality* bertujuan untuk membantu penelitian untuk mengetahui *node* yang berpotensi memiliki pengaruh signifikan dan paling berkontribusi terhadap keseluruhan graf. Untuk mencari *node* yang paling berpengaruh terhadap *node* yang lain. Dengan menggunakan *closeness centrality* dapat menunjukkan jika posisi *node* berada pada pusat graf maka *node* itu bisa disebut *node* paling berpengaruh. Algoritma Node2vec membantu untuk melihat kedekatan vektor antar *node* serta dapat melakukan klusterisasi antar *node* [10].

2. Tinjauan Pustaka

2.1. SPARQL

SPARQL *Protocol and RDF Query Language* (SPARQL) merupakan bahasa *query* yang dapat digunakan untuk memperoleh data *Resource Description Framework* (RDF) dari suatu *database*. Pada *semantic web*, RDF digunakan sebagai standar dalam melakukan pertukaran data sehingga data-data yang ada di internet dapat saling terhubung satu sama lain. SPARQL dapat digunakan pada *website* penyedia SPARQL *endpoint* seperti “*Virtuso SPARQL Query Editor*” untuk mendapatkan data RDF yang dimiliki oleh *website* tersebut [11].

2.2. Node2Vec

Node2Vec adalah algoritma *semi-supervised* yang efisien dan terukur untuk *feature learning* dalam graf. Node2Vec dapat digunakan untuk menghasilkan vektor yang merepresentasikan *node* yang ada pada graf. Node2Vec dikembangkan dari prosedur *random walk* yang fleksibel yang dapat menjelajahi *node-node* sekitarnya dalam *mode Breadth-first Sampling* dan *Depth-first Sampling* [12].

2.3. Closeness Centrality

Closeness Centrality adalah salah satu cara untuk mengukur *centrality* dalam suatu graf yang fokus terhadap seberapa dekat suatu *node* dengan *node-node* lainnya. Semakin tinggi nilai *closeness centrality* suatu *node*, maka semakin dekat pula *node* tersebut dengan *node* lainnya [3]. Nilai dari *closeness centrality* dapat dihitung dengan menggunakan rumus 1.

$$Cc(v_i) = \frac{n-1}{\sum_{j \neq i} g(v_i, v_j)} \quad (1)$$

Dimana

$g(v_i, v_j)$ = jarak antara *node* v_i dan v_j

n = jumlah *node* yang terdapat di dalam graf

3. Metodologi Penelitian

3.1. Query SPARQL

Dataset diambil dari <http://id.dbpedia.org/> dengan menggunakan SPARQL *endpoint* (*Virtuso Sparql Query Editor*) yang disediakan oleh *website* tersebut. Data yang diambil adalah seluruh data pemain sepak bola yang terdiri dari nama pemain, nama klub, dan posisi pemain, yang berada di dalam kategori “pemain sepak bola Indonesia”. *Query* yang digunakan untuk mengambil data dapat dilihat pada Kode 1.

Kode 1. SPARQL Query Untuk Mendapatkan Data Pemain Sepak Bola Indonesia

```

select DISTINCT ?nama, ?nama_klub, ?nama_posisi
where {
  ?pemain dcterms:subject
  <http://id.dbpedia.org/resource/Kategori:Pemain_sepak_bola_Indonesia>.
  ?pemain rdfs:label ?nama.
  ?pemain dbpedia-owl:position ?nama_posisi.
  ?pemain dbpprop-id:clubs ?klub.
  ?posisi rdfs:label ?nama_posisi.
  ?klub rdfs:label ?nama_klub. }

```

Selanjutnya hasil *query* SPARQL tersebut disimpan kedalam *file* berbentuk *comma seperated value* (CSV). Contoh data yang ada pada file CSV tersebut dapat dilihat pada Tabel 1.

Tabel 1. Preview Data Hasil Query Sparql

Pemain	Klub	Posisi
Ahmad Jufrianto	Sriwijaya FC	Defender
Ahmad Jufrianto	Pelita Jaya	Defender
Ahmad Jufrianto	Arema Malang	Defender
Ahmad Jufrianto	Persita	Defender
Agung Setyabudi	Persebaya Surabaya	Pemain bertahan
Agung Setyabudi	Persis Solo	Pemain bertahan
Agung Setyabudi	PSIS Semarang	Pemain bertahan
Agung Setyabudi	Arseto Solo	Pemain bertahan
Agustiar Batubara	Persebaya Surabaya	Bek
Agustiar Batubara	Persela Lamongan	Bek

Data yang berhasil didapatkan berisi total 684 baris dan tiga kolom. Kolom pemain berisi nama pemain sepak bola, kolom klub berisi nama klub yang pernah diwakili oleh pemain sepak bola, dan kolom posisi berisi posisi yang dimainkan oleh pemain sepak bola. File CSV dapat dilihat pada Zenodo [13].

Karena adanya kemiripan pada posisi-posisi pemain yang ada pada file CSV, maka dilakukan standarisasi terhadap posisi-posisi tersebut. Hasil dari standarisasi terhadap posisi-posisi pemain dapat dilihat pada Tabel 2.

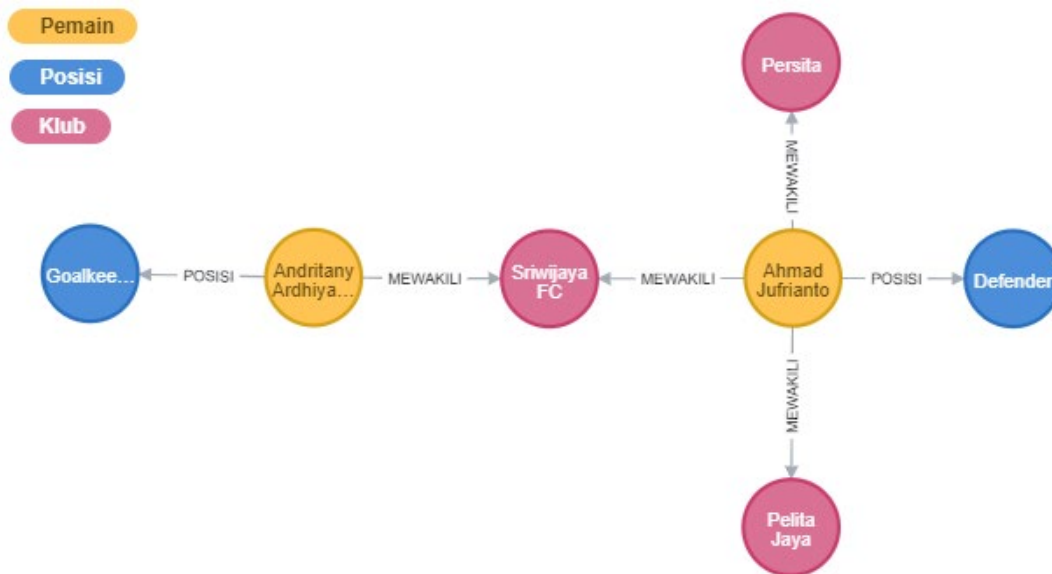
Tabel 2. Hasil Standarisasi Posisi Pemain

Posisi Lama	Posisi Baru
Kiper	Goalkeeper
Penjaga Gawang	Goalkeeper
Forward	Striker
Penyerang	Striker
Gelandang	Midfielder
Pemain tengah	Midfielder
Gelandang bertahan	Midfielder
Pemain bertahan	Defender
Bek	Defender
Pemain belakang	Defender
Pelatih Kiper	Pelatih

3.2. Pemodelan Graf

Pada tahap ini dilakukan pemodelan graf terhadap data pemain sepak bola Indonesia yang telah didapatkan pada tahap sebelumnya. Graf terdiri dari tiga *node* yaitu *node* pemain yang berisi nama dari pemain, *node* klub yang berisi nama klub dari pemain, *node* posisi yang berisi posisi bermain dari pemain. *Node-node* tersebut dihubungkan oleh dua jenis relasi yaitu relasi MEWAKILI yang menghubungkan dari *node* pemain menuju *node* klub, dan relasi POSISI yang menghubungkan dari *node* pemain menuju *node* posisi. Setiap *node* pemain dapat

mewakili lebih dari satu *node* klub. Gambar 1 menunjukkan hasil pemodelan dari graf yang telah dilakukan.



Gambar 1. Node dan Relasi Graf

3.3. Menjalankan Node2Vec

Peneliti menggunakan *library* node2vec yang ada pada python untuk menjalankan fungsi Node2Vec. Fungsi tersebut memiliki lima parameter yang harus disiapkan agar Node2Vec dapat dijalankan. Parameter-parameter tersebut antara lain: (1) Graf yang dihasilkan dari *library* NetworkX. (2) Jumlah *dimension* yang digunakan. (3) Jumlah *walk length* (jumlah *node* pada setiap *random walk*). (4) Jumlah *number of walks* (jumlah *random walk* yang dilakukan). (5) Jumlah *workers* untuk eksekusi paralel. Kode 2 untuk menjalankan Node2Vec dengan parameter *dimension* = 20, *walk_length*=16, *num_walks*=100, dan *workers*=2.

Kode 2. Kode Python Untuk Menjalankan Node2Vec

```
node2vec = Node2Vec(g, dimensions=20, walk_length=16, num_walks=100,
workers=2)
model = node2vec.fit(window=10, min_count=1)
```

Untuk lebih jelasnya dapat dilihat pada *Jupyter Notebook* Zenodo[13].

3.4. Menjalankan Algoritma Closeness Centrality

Peneliti menggunakan *library* NetworkX yang ada pada Python untuk menjalankan fungsi algoritma *closeness centrality*. Fungsi tersebut memiliki dua parameter yang harus disiapkan agar algoritma *closeness centrality* dapat dijalankan. Parameter-parameter tersebut antara lain: (1) Graf yang dihasilkan dari *library* NetworkX. (2) *Boolean* (*True/False*) jika *True* (*default*) maka akan dilakukan normalisasi dengan jumlah *node* yang terhubung pada graf.

3.5. Visualisasi

Proses pembuatan visualisasi dilakukan dengan menggunakan TSNE yang ada pada *library* sklearn.manifold dan matplotlib yang ada pada Python. TSNE berfungsi untuk melakukan pengurangan terhadap dimensi data sehingga data tersebut dapat divisualisasikan [14]. Peneliti menggunakan tiga parameter saat menjalankan TSNE, yaitu: (1) jumlah *n_components* atau dimensi yang ditargetkan, (2) angka *random_state* yang menjadi inisiasi pada algoritma TSNE, dan (3) jumlah *perplexity* atau jumlah *node* terdekat yang akan

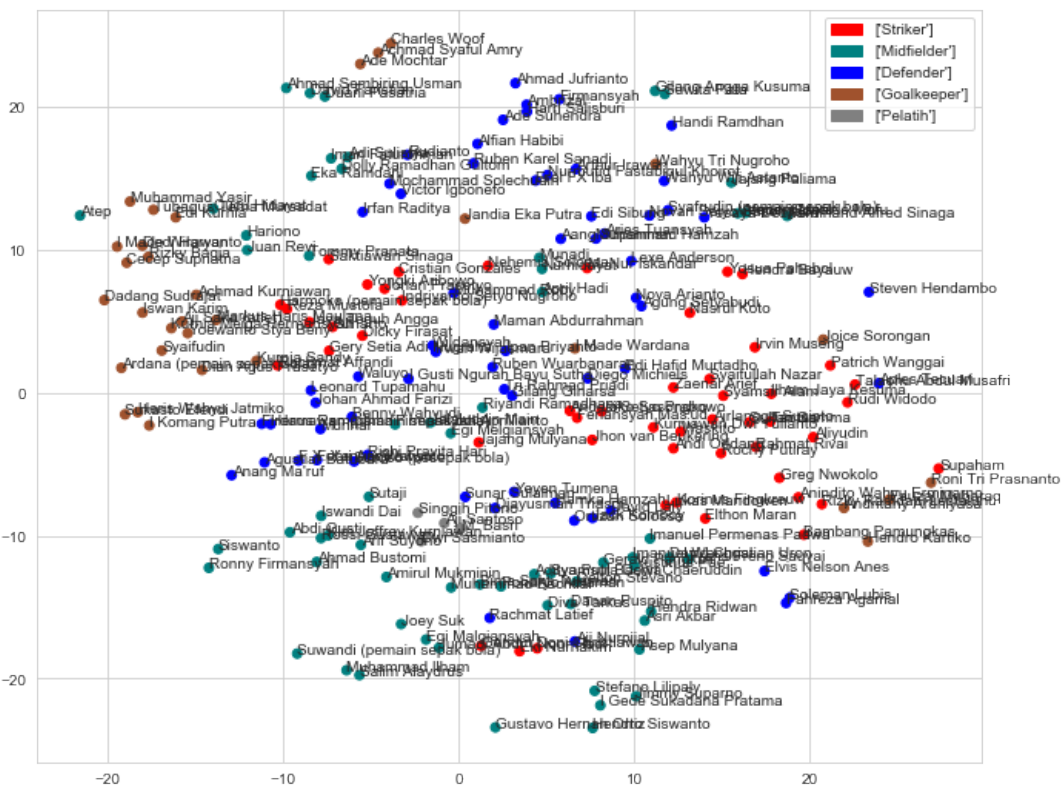
digunakan pada *learning algorithms*. Selanjutnya, hasil TSNE akan divisualisasikan menjadi *scatter plot* dengan menggunakan *matplotlib*.

4. Hasil dan Diskusi

4.1. Hasil Node2Vec

Node2Vec mengubah matrik yang ada pada graf menjadi vektor yang memiliki dimensi lebih kecil. Setelah pemodelan Node2Vec berhasil dilakukan, langkah selanjutnya adalah membuat visualisasi terhadap model Node2Vec. *Library* *sklearn.manifold* digunakan untuk mendapatkan fungsi TSNE yang berguna untuk mengurangi dimensi data menjadi dua dimensi. Hasil dari TSNE divisualisasi menjadi *scatter plot* dengan menggunakan *library matplotlib*.

Pada Gambar 2 dapat dilihat hasil dari visualisasi pemain sepak bola dilihat dari posisi bermain-nya. Melalui visualisasi tersebut dapat dilihat pemain yang berwarna merah adalah *striker*, pemain yang berwarna hijau adalah *midfielder*, pemain yang berwarna biru adalah *defender*, pemain yang berwarna coklat adalah *goalkeeper*, dan pemain yang berwarna abu-abu adalah pelatih.



Gambar 2. Visualisasi Pemain Sepak Bola Dilihat Dari Posisi

Hasil visualisasi menunjukkan posisi *Defender* yang merupakan posisi dengan pemain terbanyak terlihat lebih menyebar dibandingkan dengan posisi-posisi lainnya. Posisi *Goalkeeper* cenderung berkumpul disebelah kiri, posisi *Midfielder* cenderung berkumpul dibagian bawah, dan posisi *Striker* cenderung berkumpul disebelah kanan. Pemain yang tidak ikut berkumpul dengan posisi-nya bisa disebabkan oleh adanya pemain lain dengan posisi yang berbeda yang mewakili klub yang sama dengan pemain tersebut. Contohnya bisa dilihat pada *node* pemain “Ardana (pemain sepak bola)” yang hanya mewakili satu *node* klub “PS Bengkulu”, dan tidak ada pemain lain yang mewakili *node* klub tersebut. Posisi *node* pemain “Ardana (pemain sepak bola)” berada di sebelah kiri dan berkumpul dengan *node-node* dengan posisi *Goalkeeper* lainnya.

4.2. Hasil Algoritma Closeness Centrality

Graf yang akan dihitung nilai *closeness centrality*-nya, memiliki nilai *density* sebesar 0,013652344641797178. Nilai *closeness centrality* dapat dijadikan sebagai patokan berpengaruh atau tidaknya suatu *node*, karena semakin tinggi nilai *closeness centrality* maka semakin dekat *node* tersebut dengan *node-node* lainnya. Melalui algoritma *closeness centrality* didapatkan tiga *node* yang paling berpengaruh serta tiga *node* yang paling tidak berpengaruh. Ketiga *node* paling berpengaruh didapatkan dari tiga *node* dengan nilai *closeness centrality* tertinggi yang dapat dilihat pada Tabel 3. Ketiga *node* paling tidak berpengaruh dari tiga *node* dengan nilai *closeness centrality* terendah yang dapat dilihat pada Tabel 4.

Tabel 3. Tiga Pemain Paling Berpengaruh

Node	Closeness Centrality
Rachmat Affandi	0,3507
Gendut Doni Christiawan	0,35
Hendra Ridwan	0,3345

Pada Tabel 3 dapat dilihat bahwa ketiga *node* dengan nilai *closeness centrality* tertinggi adalah *node* “Rachmat Affandi”, “Gendut Doni Christiawan”, dan “Hendra Ridwan”. Ketiga pemain tersebut berada pada banyak klub sepak bola yang juga memiliki pemain-pemain yang terhubung dengan ketiga pemain tersebut. Selain itu posisi ketiga pemain tersebut adalah *striker* dimana *node striker* memiliki nilai *closeness centrality* lebih tinggi daripada *node “Defender”*.

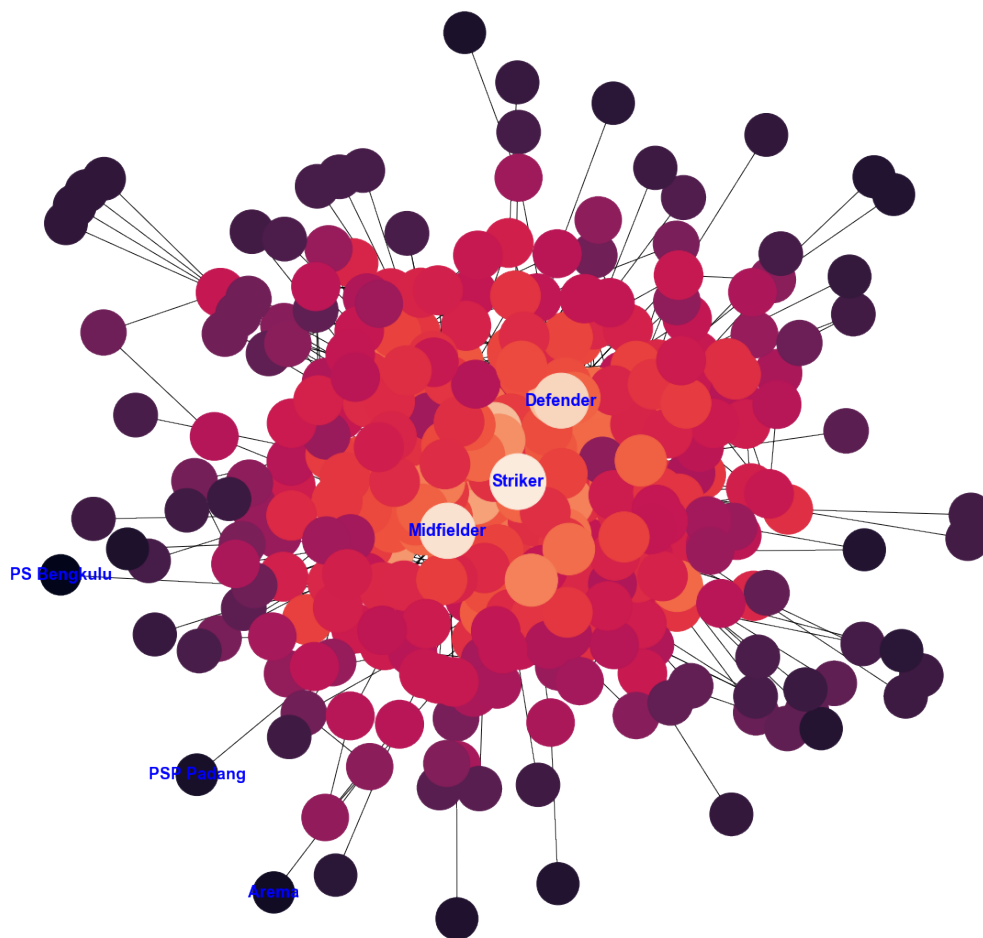
Tabel 4. Tiga Node Paling Tidak Berpengaruh

Node	Tipe	Closeness Centrality
PS Bengkulu	Klub	0,19770617149098854
Arema	Klub	0,20533182076006806
PSP Padang	Klub	0,20864553314121037

Pada Tabel 4 dapat dilihat bahwa ketiga *node* dengan nilai *closeness centrality* terendah adalah *node* “PS Bengkulu”, “Arema”, dan “PSP Padang”. Ketiga *node* tersebut merupakan *node* bertipe klub dan hanya memiliki satu *degree* saja. Akibatnya ketiga *node* tersebut memiliki jumlah jarak antar *node* yang sangat besar yang menyebabkan nilai *closeness centrality*-nya sangat kecil.

Library matplotlib digunakan dalam melakukan visualisasi terhadap hasil algoritma *closeness centrality*. *Node-node* divisualisasikan dengan ukuran dan warna yang berbeda sesuai dengan jumlah *degree* dan nilai *closeness centrality*-nya. Semakin besar nilai *closeness centrality* suatu *node*, maka akan semakin besar ukuran *node*. Semakin besar jumlah *degree* suatu *node*, maka akan semakin cerah warna *node* tersebut. Gambar 3 merupakan hasil visualisasi dari graf berdasarkan nilai *closeness centrality*-nya.

Pada Gambar 3 dapat dilihat bahwa *node* “Striker”, “Midfielder”, dan “Defender” yang merupakan pemiliki nilai *closeness centrality* tertinggi memiliki ukuran *node* yang besar. Selain itu, ketiga *node* tersebut juga berwarna cerah dan berada di tengah graf karena banyak yang terhubung dengan ketiga *node* tersebut. Sedangkan *node* “PS Bengkulu”, “Arema”, dan “PSP Padang” yang merupakan pemilik nilai *closeness centrality* terendah memiliki ukuran *node* yang kecil, warna yang gelap, dan berada di pinggir graf karena hanya memiliki satu *degree*. Pada visualisasi ini juga dapat terlihat jelas bahwa posisi *node* “Striker” lebih *central* daripada *node* “Defender”, yang menyebabkan nilai *closeness centrality* dari *node* “Striker” menjadi lebih besar.



Gambar 3. Visualisasi Hasil Algoritma Closeness Centrality

5. Kesimpulan

Penelitian ini mengimplementasikan algoritma Node2Vec dan algoritma *closeness centrality* pada *dataset* pemain sepak bola Indonesia. Node2Vec dijalankan untuk mengubah *node-node* yang ada pada graf menjadi vektor. Setelah Node2Vec berhasil dijalankan, dilakukan visualisasi dengan menggunakan TSNE dan matplotlib untuk melihat hasilnya. Selanjutnya algoritma *closeness centrality* dijalankan untuk melihat *node* apa yang memiliki kedekatan paling tinggi dengan *node-node* lainnya. Dengan melihat nilai *closeness centrality* dari tiap *node*, maka kita dapat mengidentifikasi *node* apa yang memiliki pengaruh paling besar terhadap *node-node* lainnya pada suatu graf. Setelah nilai *closeness centrality* didapatkan, dilakukan visualisasi dengan menggunakan matplotlib untuk melihat hasilnya.

Kesimpulan dari penelitian ini adalah pemain *defender* tersebar, sedangkan kiper terkumpul pada visualisasi Node2Vec. Dengan melihat hasil *closeness centrality*, pemain *striker* memiliki nilai lebih tinggi daripada posisi pemain lainnya.

Referensi

- [1] M. Yani, "Pemanfaatan linked data untuk meningkatkan kualitas information knowledge pada resources game online," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 6, no. 2, pp. 168–173, May 2017.
- [2] N. A. Rakhmawati, *Semantic Web dan Linked data*. Yogyakarta, Indonesia: SiBuku, 2015.
- [3] F. Y. Pratama, "Simulasi jejaring jalan Kota Pontianak dengan betweenness centrality dan

- degree centrality,” *J. TIN Univ. Tanjungpura*, vol. 2, no. 1, pp. 1-6, 2018.
- [4] B. Susanto, H. Lina, and A. R. Chrismanto, “Penerapan social network analysis dalam penentuan centrality studi kasus social network Twitter,” *J. Inform.*, vol. 8, no. 1, Apr. 2012.
- [5] N. Insani, “Pemanfaatan networkx untuk mengeksplorasi dan menganalisa jaringan beserta sifat/karakteristiknya,” in *Seminar Nasional Matematika dan Pendidikan Matematika*, Nov. 2012.
- [6] A. Rochiyat and A. Wibowo, “Analisis aktor berpengaruh dan aktor populer dengan metode Degree Centrality dan follower rank pada tagar Twitter ‘#gejayanmemanggil,’” *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 6, no. 2, Dec. 2019, doi: 10.35957/jatisi.v6i2.187.
- [7] K. Mufidah, N. Syahputra, and N. A. Rakhmawati, “Analisis aktor populer dan sutradara berpengaruh berdasarkan data dbpedia menggunakan algoritma Closeness Centrality dan Node2vec,” *UNIKOM*, vol. 18, no. 1, pp. 37–48, 2020.
- [8] A. Rahman, M. A. Naufal, and N. A. Rakhmawati, “Analisis pengaruh wilayah dan partai politik terhadap hubungan para aktivis di Indonesia menggunakan algoritma Degree Centrality,” *J. Inform. Upgris*, vol. 6, no. 1, Jul. 2020, doi: 10.26877/jiu.v6i1.6092.
- [9] N. A. Rakhmawati, R. Alfarhizi, and I. Hafidz, “Penerapan social network analysis dengan menggunakan metode Sociomatrix pada akun Instagram siswa SMA di Surabaya,” *SISTEMASI*, vol. 9, no. 2, May 2020, doi: 10.32520/stmsi.v9i2.807.
- [10] Z. Shen, F. Chen, L. Yang, and J. Wu, “Node2vec representation for clustering journals and as a possible measure of diversity,” *J. Data Inf. Sci.*, vol. 4, no. 2, pp. 79–92, Jun. 2019.
- [11] *SPARQL 1.1 Overview. W3C Recommendation 21 March 2013*, W3C, Mar. 2013. [Online]. Available: <https://www.w3.org/TR/sparql11-overview/>
- [12] A. Grover and J. Leskovec, “Node2vec: scalable feature learning for networks,” in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Aug. 2016, pp. 855–864, doi: <http://dx.doi.org/10.1145/2939672.2939754>.
- [13] A. N. R., “supep/FPTekWeb: implementation of Node2Vec and Closeness Centrality on Indonesian football players graph,” May 26, 2020. Distributed by Zenodo. doi: 10.5281/zenodo.3851899.
- [14] L. van der Maaten and G. Hinton, “Visualizing data using t-SNE,” *J. Mach. Learn. Res.*, vol. 9, pp. 2579–2605, Nov. 2008.