

Eksplorasi *Bot Farms* terhadap Algoritma Media Sosial: Analisis Graf Berbobot sebagai Strategi Mitigasi Penyebaran Hoaks

Rafen Max Alessandro - 13523031^{1,2}

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

¹13523031@std.stei.itb.ac.id, ²rafen.max@gmail.com

Abstract— Graf berbobot menyediakan peran bagi algoritma media sosial modern untuk memodelkan hubungan dan mendistribusikan konten berdasarkan interaksi antar pengguna. Namun, algoritma ini dapat disalahgunakan untuk tujuan manipulasi informasi, seperti tindakan penyebaran hoaks yang terorganisir. Dalam konteks keamanan siber, *bot farms* memainkan peran signifikan dalam memanfaatkan graf berbobot untuk mempercepat penyebaran informasi palsu, menciptakan ilusi relevansi dan popularitas pada konten hoaks, sehingga algoritma media sosial memprioritaskannya dalam distribusi konten. Sebagai upaya dalam penanganan penyebaran hoaks, beberapa pendekatan seperti deteksi bot berbasis pola graf, diversifikasi konten dengan metode *epsilon-greedy*, pemberian prioritas terhadap akun tepercaya, dan literasi digital menjadi strategi mitigasi yang mampu meminimalisir terbentuknya fenomena *filter bubble*, *echo chamber*, dan polarisasi opini pada media sosial, menekan frekuensi penyebaran hoaks berbasis algoritma media sosial secara signifikan.

Keywords—algoritma media sosial, *bot farms*, graf berbobot, penyebaran hoaks.

I. PENDAHULUAN

Media sosial telah menjadi komponen krusial dalam kehidupan sehari-hari masyarakat di era digital, menjadi wadah bagi individu dan kelompok untuk berinteraksi, berbagi informasi, dan membangun jaringan secara daring. Media sosial yang telah beredar di masyarakat, seperti Instagram, Facebook, Twitter, dan lain sebagainya, dirancang untuk memfasilitasi komunikasi langsung dan distribusi konten secara luas, mencakup teks, gambar, video, dan berbagai macam format konten lainnya. Keberlangsungan media sosial untuk memberikan pengalaman menyenangkan bagi penggunaannya didukung oleh algoritma kompleks untuk menyaring dan memprioritaskan konten yang ditampilkan pada beranda untuk setiap pengguna. Algoritma media sosial modern didasarkan terhadap interaksi yang terjadi antar pengguna, seperti jumlah *like*, komentar, *share*, dan tingkat keterlibatan antar akun. Pendekatan algoritma berbasis interaksi dirancang untuk meningkatkan personalisasi dengan menyesuaikan konten yang ditampilkan sesuai dengan preferensi pengguna, sehingga pengalaman pengguna dalam menggunakan media sosial meningkat dalam taraf yang signifikan.

Namun, pengaplikasian secara nyata algoritma media sosial

berbasis interaksi memunculkan tantangan baru yang mendorong penyebaran hoaks di masyarakat. Tantangan ini muncul sebagai dampak dari fenomena *filter bubble* dan *echo chamber* yang timbul dari implementasi algoritma berbasis interaksi. Kedua fenomena tersebut memperkuat penyebaran hoaks melalui media sosial dengan memengaruhi pengguna untuk hanya menerima dan mempercayai informasi yang selaras dengan pandangan yang dimiliki tanpa memverifikasi lebih lanjut. Menggunakan fenomena ini, oknum tidak bertanggung jawab dapat memicu terjadinya polarisasi opini dengan memanipulasi opini publik, mengarahkan orientasi perilaku massa, hingga mengganggu stabilitas sosial-politik masyarakat. Dampak-dampak tersebut dapat diraih melalui eksploitasi *bot farms* untuk memanipulasi algoritma media sosial berbasis interaksi dan menyebarkan hoaks kepada masyarakat.

Bot farms memanipulasi algoritma media sosial berbasis interaksi dengan memanfaatkan pola-pola yang digunakan algoritma dalam menentukan konten yang ditampilkan kepada pengguna. *Bot farms* merekayasa interaksi intensif antar akun bot, atau bahkan dengan akun pengguna, agar algoritma menganggap konten yang disebarluaskan oleh bot merupakan konten otentik yang layak untuk didistribusikan secara luas. *Bot farms* juga merekayasa *cluster* dengan hubungan internal yang kuat untuk membentuk pseudo-komunitas mengenai suatu konten, sehingga kredibilitas hoaks meningkat untuk disebarluaskan oleh algoritma. *Bot farms* dapat dengan mudah menyiarkan hoaks kepada pengguna media sosial dengan memanfaatkan algoritma berbasis interaksi, meningkatkan risiko terjadinya polarisasi opini di masyarakat.

Makalah ini melakukan pendekatan menggunakan graf berbobot untuk mengelaborasi algoritma media sosial berbasis interaksi, membentuk studi kasus yang memodelkan algoritma, serta menguraikan cara bagaimana *bot farms* memanipulasi algoritma. Melalui analisis yang dilakukan, dibentuk strategi-strategi yang dapat diaplikasikan sebagai upaya melakukan mitigasi terhadap peran *bot farms* dalam algoritma media sosial berbasis interaksi sebagai cara untuk menjaga integritas ruang digital yang menjadikan kebenaran informasi sebagai pilar utama.

II. LANDASAN TEORI

A. Algoritma Media Sosial



Gambar 1. Media Sosial

Sumber: <https://uici.ac.id/ini-7-media-sosial-paling-banyak-digunakan-di-indonesia/>

Media sosial adalah platform digital yang dirancang untuk memfasilitasi komunikasi, distribusi informasi, dan interaksi sosial antar individu dan kelompok secara daring. Diawali dengan hadirnya platform seperti Friendster (2002) dan MySpace (2003), media sosial telah berkembang pesat menjadi berbagai bentuk platform populer yang tidak asing bagi masyarakat digital, seperti Facebook, Instagram, Twitter, dan TikTok. Media sosial awalnya berperan sebagai sarana koneksi antar pengguna, tetapi telah berkembang dan beralih fungsi sebagai ruang untuk distribusi informasi, seperti iklan, hiburan, dan ekspresi sosial.

Pengalaman yang didapatkan pengguna dalam menggunakan media sosial dipengaruhi oleh algoritma yang mendasari operasinya. Algoritma media sosial adalah seperangkat aturan berbasis data dan model matematika yang dirancang untuk menyaring, memprioritaskan, dan menampilkan konten yang dianggap paling relevan bagi pengguna. Algoritma media sosial awalnya berbasis temporal dan menampilkan konten secara kronologis, sehingga pengguna akan menerima kiriman konten terbaru terlebih dahulu. Pendekatan algoritma berbasis temporal memberikan pengalaman yang sederhana dan kontekstual, tetapi menghadapi keterbatasan akibat meningkatnya jumlah pengguna dan volume konten seiring waktu. Algoritma sulit untuk menemukan konten yang relevan dalam tumpukan informasi yang terus bertambah secara pesat.

Untuk mengatasi tantangan ini, algoritma media sosial kini berkembang menjadi berbasis interaksi. Dalam algoritma yang lebih modern, konten tidak hanya diurutkan berdasarkan waktu, tetapi juga diurutkan berdasarkan penilaian terhadap tingkat keterlibatan pengguna, biasanya dinyatakan dalam bentuk *like*, komentar, *share*, atau durasi waktu yang dihabiskan untuk melihat konten. Selain mengatasi tantangan yang dihadapi oleh algoritma berbasis temporal, algoritma ini meningkatkan personalisasi untuk setiap pengguna. Konten yang ditampilkan oleh algoritma disesuaikan dengan tingkatan preferensi yang pengguna miliki terhadap konten tertentu, meningkatkan pengalaman pengguna dalam menggunakan media sosial.

Namun, implementasi algoritma media sosial berbasis interaksi membawa konsekuensi bagi masyarakat digital. Dengan memprioritaskan keterlibatan pengguna, algoritma cenderung memberikan eksposur terhadap konten sensasional yang lebih menarik perhatian pengguna, mengesampingkan kebenaran dari konten tersebut. Tantangan lain seperti fenomena

filter bubble dan *echo chamber* yang menggambarkan keadaan pengguna hanya terpapar konten sesuai dengan pandangan sendiri hingga berujung kepada penyebaran hoaks dan polarisasi opini menjadi ancaman yang mengganggu integritas ruang digital dan dapat berakibat fatal bagi dinamika masyarakat digital.

B. Graf

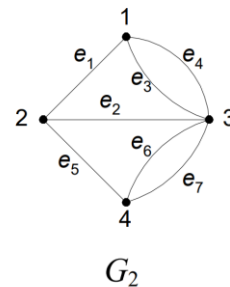
Graf merupakan representasi kumpulan objek diskrit dan hubungan yang dimiliki antara objek-objek tersebut. Leonhard Euler pertama kali menggunakan representasi graf pada tahun 1736 dalam persoalan jembatan Königsberg. Menggunakan pembuktian dengan kontradiksi, Euler mendasari teori-teori graf yang digunakan hingga saat ini [1].

Secara matematis, graf G didefinisikan dalam bentuk $G = (V, E)$ dengan V merupakan himpunan simpul (*vertices*) yang dipastikan bukan himpunan kosong, dan E merupakan himpunan sisi (*edges*) yang berperan sebagai hubungan antara dua simpul. Simpul pada sebuah graf umumnya dinyatakan dengan huruf atau angka, sedangkan sisi dinyatakan dalam bentuk (a, b) atau e , dengan a dan b melambangkan dua simpul yang dihubungkan oleh sisi tersebut. Dua sisi yang menghubungkan dua simpul yang sama disebut dengan istilah sisi ganda (*multiple edges*), dan sisi yang berawal dan berakhir pada simpul yang sama disebut dengan istilah gelang (*loop*). Sebagai contoh, pada gambar 2, graf G_2 adalah graf dengan definisi matematis untuk simpul dan sisi sebagai berikut [1].

$$V = \{1, 2, 3, 4\}$$

$$E = \{(1, 2), (2, 3), (1, 3), (1, 3), (2, 4), (3, 4), (3, 4)\} \text{ atau}$$

$$E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7\}$$



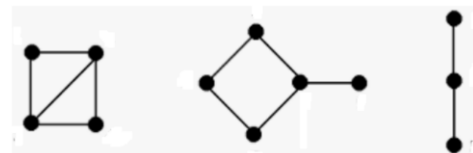
Gambar 2. Graf G_2

Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

Berdasarkan eksistensi sisi ganda dan / atau gelang pada suatu graf, graf digolongkan menjadi dua jenis:

1. Graf sederhana (*simple graph*)

Graf sederhana adalah graf yang tidak mengandung sisi ganda maupun gelang [1].

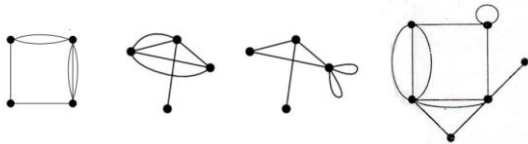


Gambar 3. Graf Sederhana (*Simple Graph*)

Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

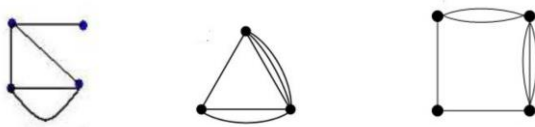
2. Graf tak-sederhana (*unsimple graph*)

Graf yang mengandung sisi ganda atau gelang dinamakan graf tak-sederhana [1].

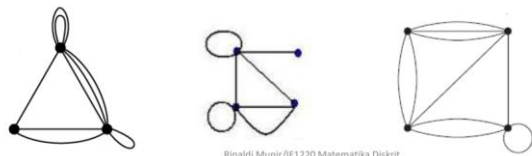


Gambar 4. Graf Tak-sederhana (*Unsimple Graph*)
Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

Lebih lanjut, graf tak-sederhana dikategorikan menjadi dua, graf ganda dan graf semu. Sesuai namanya, graf ganda (*multi-graph*) adalah graf yang memiliki sisi ganda dalam himpunan sisi graf, sedangkan graf semu (*pseudo-graph*) adalah graf yang memiliki sisi gelang dalam himpunan sisi graf [1].



Gambar 5. Graf Ganda (*Multi-graph*)
Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

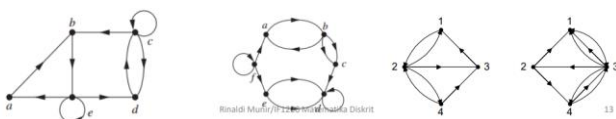


Gambar 6. Graf Semu (*Pseudo-graph*)
Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

Berdasarkan orientasi arah pada sisi, graf dibedakan menjadi 2 jenis, yaitu graf tak-berarah (*undirected graph*) dan graf berarah (*directed graph*). Sisi pada graf tak-berarah tidak memiliki orientasi arah, sehingga sisi hanya menggambarkan hubungan antara dua buah simpul. Sedangkan sisi pada graf berarah memiliki orientasi arah, keluar / menjauhi satu simpul dan masuk / mengarah ke simpul yang lain. Sisi pada graf berarah menggambarkan hubungan yang lebih kompleks antara dua buah simpul [1].



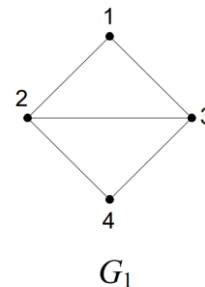
Gambar 7. Graf Tak-berarah (*Undirected Graph*)
Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>



Gambar 8. Graf Berarah (*Directed Graph*)
Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

Jenis	Sisi	Sisi Ganda	Gelang
Graf sederhana	Tak-berarah	Tidak	Tidak
Graf ganda	Tak-berarah	Ya	Tidak
Graf semu	Tak-berarah	Ya	Ya
Graf berarah	Berarah	Tidak	Ya
Graf ganda berarah	Berarah	Ya	Ya

Sebagai sebuah konsep matematis, terdapat banyak terminologi yang mendeskripsikan komponen relevan sebuah graf. Salah satu terminologi yang mendefinisikan komponen esensial dalam graf adalah ketetanggaan (*adjacency*). Ketetanggaan menggambarkan hubungan atau relasi yang dimiliki antara simpul-simpul dalam sebuah graf. Dua buah simpul yang terhubung melalui sisi disebut dengan istilah simpul ketetanggaan [1]. Terminologi ketetanggaan umumnya digunakan untuk mendeskripsikan struktur graf, misalnya dalam persoalan pencarian rute terpendek atau untuk menyatakan hubungan dalam suatu jaringan komputer. Pada gambar 9, simpul 1 graf G_1 dinyatakan bertetangga dengan simpul 2 dan 3, tetapi tidak bertetangga dengan simpul 4 karena tidak terdapat sisi yang menghubungkan kedua simpul tersebut.

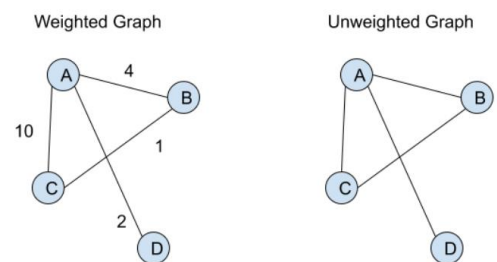


Gambar 9. Graf G_1

Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

C. Graf Berbobot

Graf berbobot (*weighted graph*) adalah klasifikasi graf dengan bobot numerik untuk setiap sisi yang dimiliki graf [1]. Bobot yang dimiliki sisi merepresentasikan atribut yang dimiliki oleh hubungan antara dua simpul bertetangga, misalnya jarak, biaya, waktu, atau intensitas hubungan. Karakteristik bobot untuk sisi yang dimiliki oleh graf berbobot menjadi model yang baik dan fleksibel untuk digunakan dalam berbagai aplikasi praktis untuk pengaplikasian yang didasari terhadap sisi dalam graf.

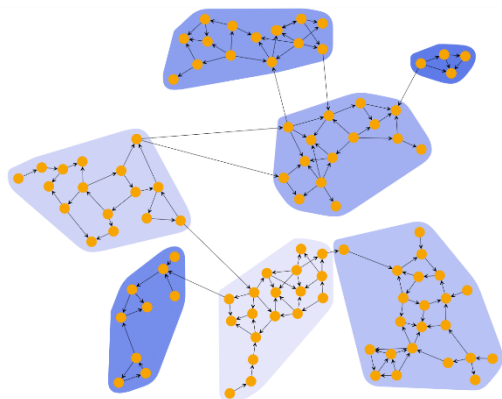


Gambar X. Graf Berbobot (*Weighted Graph*) dan Graf Tak-berbobot (*Unweighted Graph*)

Sumber: <https://informatika.stei.itb.ac.id/~rinaldi.munir/>

Tabel I. Jenis-jenis Graf

Graf berbobot dapat dimanfaatkan sebagai dasar dalam pembentukan *cluster* untuk simpul dalam graf. *Cluster* merujuk kepada kumpulan simpul yang memiliki hubungan internal lebih erat dibandingkan hubungan eksternal dengan simpul di luar *cluster*. Keeratan hubungan internal pada sebuah *cluster* dicirikan dengan bobot sisi yang tinggi, merepresentasikan intensitas hubungan antar simpul dalam *cluster*. Menggunakan graf berbobot, *cluster* dapat diidentifikasi menggunakan algoritma seperti Louvain, K-means, atau Spectral Clustering yang mempertimbangkan nilai dari bobot setiap sisi dalam graf untuk menentukan *cluster* yang terbentuk dalam sebuah graf. *Cluster* dapat dimanfaatkan sebagai sarana untuk memahami struktur jaringan dan interaksi yang terjadi dalam suatu graf [2].



Gambar 11. Cluster pada Graf

Sumber: <https://www.yworks.com/pages/clustering-graphs-and-networks>

D. Bot

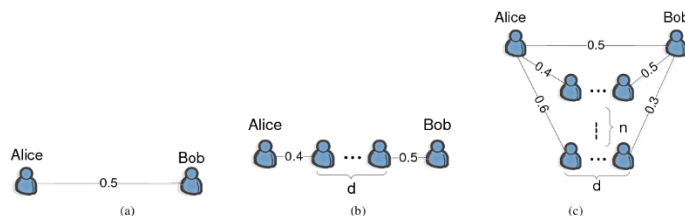
Bot adalah program perangkat lunak yang dibentuk untuk menjalankan tugas otomatis dalam suatu lingkungan digital. Bot dapat beroperasi tanpa atau dengan interaksi minimal dari manusia, sehingga bot umumnya digunakan untuk melakukan tugas berulang dengan efisiensi tinggi. Salah satu contoh pertama dari implementasi bot adalah ELIZA pada tahun 1966, sebuah program yang dirancang untuk mensimulasikan percakapan manusia. Seiring dengan berjalannya waktu, bot berkembang membentuk entitas yang lebih kompleks dengan kemampuan yang lebih mumpuni, mencakup pemrosesan bahasa alami, analisis data, dan interaksi di berbagai platform digital, seperti media sosial.

Di era digital, bot digunakan secara modern dalam berbagai macam aplikasi praktis, seperti asisten virtual, layanan pelanggan otomatis, analisis data secara langsung, dan pengendalian perangkat lunak. Bot juga memberikan perannya dalam media sosial dengan melakukan moderasi konten, memberikan notifikasi, dan mengelola akun untuk memberikan pengalaman memuaskan berselancar di media sosial. Namun, di bawah tangan yang tidak bertanggung jawab, bot dapat disalahgunakan untuk menjalankan tindakan yang mengganggu integritas dunia digital, seperti menyebarkan disinformasi, memanipulasi opini publik, atau meningkatkan metrik palsu melalui interaksi otomatis antara bot.

III. PEMBAHASAN

Algoritma media sosial modern didasarkan terhadap basis interaksi, mempelajari preferensi pengguna berdasarkan pola

keterlibatan yang ditampilkan dalam dinamika penggunaan media sosial oleh pengguna, meliputi *like*, komentar, *share*, dan lain sebagainya. Interaksi atau pola keterlibatan yang ditunjukkan oleh penggunaan dapat dinumerasikan sebagai sebuah angka, dan digunakan sebagai bobot sisi pada sebuah graf berbobot. Graf berbobot dapat memodelkan dengan baik hubungan antara pengguna dalam suatu media sosial berdasarkan intensitas interaksi, dengan simpul merepresentasikan pengguna dan sisi merepresentasikan hubungan yang memiliki bobot intensitas interaksi yang terjadi [3], seperti yang ditunjukkan pada gambar 12.



Gambar 12. Visualisasi Graf Berbobot dalam Algoritma Media Sosial
 Sumber: Lin, X., Shang, T., & Liu, J. (2014). An estimation method for relationship strength in weighted social network graphs. *Journal of Computer and Communications*, 2(04), 82.

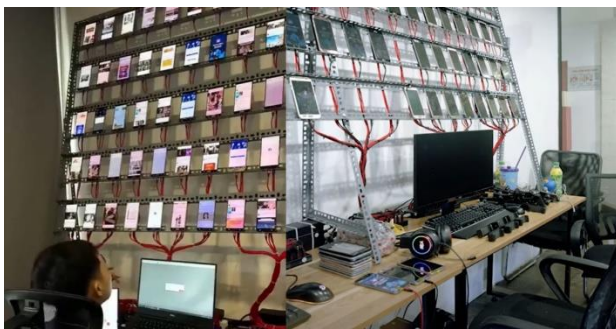
Algoritma media sosial berbasis interaksi memanfaatkan graf berbobot untuk menentukan prioritas konten. Kiriman dari akun dengan hubungan berbobot besar terhadap suatu pengguna lebih mungkin untuk ditampilkan kepada pengguna tersebut. Dengan menggunakan intensitas interaksi sebagai dasar penampihan konten, algoritma ini menjawab permasalahan penggunaan algoritma berbasis temporal terhadap peningkatan jumlah pengguna dan volume konten, sekaligus memberikan pengalaman berselancar di media sosial yang lebih baik dan relevan bagi pengguna.

Namun, algoritma berbasis interaksi turut berkontribusi dalam memperkuat terbentuknya fenomena *filter bubble* dan *echo chamber*. Fenomena *filter bubble* adalah dampak akibat penyaringan konten yang ditampilkan kepada pengguna oleh algoritma media sosial terlalu menyesuaikan dengan pandangan pengguna secara berlebihan, bahkan sama sekali mengabaikan konten yang berlawanan pandangan. Secara teori, kondisi tersebut tidak berdampak negatif dan memberikan pengalaman yang baik bagi pengguna. Namun, jika pengguna memiliki pandangan yang salah, algoritma justru memberikan ancaman dalam tingkatan yang substansial dengan tidak menampilkan konten yang berlawanan pandangan, membentuk *echo chamber* bagi pengguna mengenai pandangannya yang salah.

Echo chamber menggambarkan ekosistem sosial, luring atau daring, dengan kondisi individu hanya berinteraksi dengan individu atau kelompok yang memiliki pandangan serupa. *Echo chamber* menciptakan ruang informasi yang homogen dan memperkuat keyakinan yang sudah ada, tanpa memedulikan semantik dari keyakinan tersebut. Algoritma media sosial berbasis interaksi sangat rentan untuk mengurung pengguna ke dalam *echo chamber* yang tidak terekspos terhadap perspektif alternatif. Fenomena ini dapat dimodelkan dengan konsep *cluster* pada suatu graf berbobot. Algoritma media sosial yang hanya memberikan konten sependangan dengan pengguna memperkuat polarisasi opini di kalangan pemilih dalam suatu

cluster. Dengan sudut pandang yang terbatas, penyebaran disinformasi semakin menjadi-jadi dan menurunkan kemampuan masyarakat untuk berdiskusi secara sehat dan terbuka.

Selain itu, melalui eksploitasi algoritma media sosial yang menunjang fenomena *filter bubble* dan *echo chamber*, pihak tidak bertanggung jawab memperparah terjadinya polarisasi opini dengan melakukan eksploitasi dan manipulasi algoritma menggunakan *bot farms*. *Bot farms* adalah istilah yang mengacu kepada pengelolaan secara kolektif jaringan terorganisir dari akun-akun bot dengan tujuan tertentu [5]. Pada media sosial, tujuan dari disusunnya suatu *bot farms* umumnya meliputi manipulasi informasi, penyebaran konten tertentu, atau meningkatkan keterlibatan palsu. *Bot farms* memanipulasi algoritma berbasis interaksi pada media sosial dengan menentukan konten yang akan ditampilkan pengguna. *Bot farms* kemudian membentuk aktivitas interaksi yang intens di antara akun bot mengenai konten tersebut, misalnya mengarahkan ribuan akun untuk memberikan *like*, komentar, dan *share* konten tertentu dalam waktu singkat untuk memberikan ilusi bahwa konten tersebut populer dan relevan. Akibatnya, terbentuk *cluster* dengan hubungan internal yang sangat kuat antara pengguna dalam *cluster* tersebut yang mengarah kepada anggapan bagi algoritma media sosial bahwa konten mendukung terjadinya interaksi tinggi antar pengguna. Algoritma kemudian memberikan prioritas lebih tinggi dalam distribusi konten sehingga menjangkau pengguna nyata secara menyeluruh, tanpa memerhatikan nilai semantik dari konten yang disebarluaskan.



Gambar 13. Eksploitasi *Bot Farms* dengan Tujuan Tertentu
Sumber: <https://wakeup.sg/phone-bot-farms-social-media-fame/>

IV. STUDI KASUS

Berdasarkan landasan teori dan pembahasan yang disampaikan mengenai algoritma media sosial berbasis interaksi, graf berbobot, dan *bot farms*, dapat dibentuk suatu pemodelan graf berbobot yang menggambarkan bagaimana algoritma media sosial berbasis interaksi bekerja. Pemodelan juga menggambarkan tahapan *bot farms* berperan dalam memanipulasi algoritma untuk menyebarkan konten kepada pengguna media sosial. Pada model graf berbobot yang dibentuk menggunakan bahasa pemrograman Python, simpul merepresentasikan akun media sosial dan bobot pada sisi antara dua buah simpul menunjukkan tingkatan frekuensi atau intensitas interaksi antara akun yang terhubung.

Untuk mengilustrasikan bagaimana *bot farms* bekerja, dapat ditambahkan banyak interaksi otomatis antara akun bot yang

satu dengan yang lainnya. Rekayasa interaksi antara akun bot akan membentuk *cluster* dengan hubungan internal tinggi, memanipulasi algoritma untuk menganggap konten yang didiskusikan dalam *cluster* akun bot merupakan konten relevan dan menyebarkan ke pengguna.

1. Implementasi graf berbobot sebagai model algoritma media sosial berbasis interaksi

Implementasi graf berbobot pada Python dilakukan menggunakan library NetworkX yang mampu menciptakan, menganalisis, dan memvisualisasikan struktur graf dan jaringan. Sebagai contoh studi kasus, dibentuk 5 buah akun media sosial dengan nama akun dan interaksi antar akun sebagai berikut.

```
# implementasi graf
G = nx.Graph()

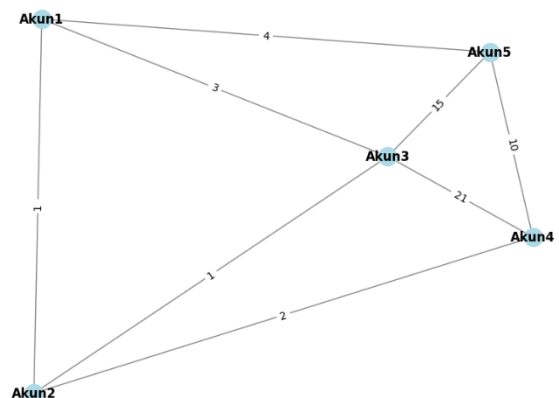
# menambahkan akun sebagai simpul
nodes = ['Akun1', 'Akun2', 'Akun3', 'Akun4', 'Akun5']
G.add_nodes_from(nodes)

# menambahkan intensitas interaksi sebagai sisi berbobot
edges = [
    ('Akun1', 'Akun2', 1), ('Akun1', 'Akun3', 3),
    ('Akun1', 'Akun5', 4), ('Akun2', 'Akun3', 1),
    ('Akun2', 'Akun4', 2), ('Akun3', 'Akun4', 21),
    ('Akun3', 'Akun5', 15), ('Akun4', 'Akun5', 10)
]
G.add_weighted_edges_from(edges)
```

Gambar 14. Penetapan Akun sebagai Simpul dan Interaksi sebagai Sisi Berbobot

Sumber: dokumentasi pribadi

Menggunakan simpul dan sisi yang telah ditentukan, graf berbobot kemudian diimplementasikan untuk menghasilkan visualisasi graf sebagai berikut.



Gambar 15. Visualisasi Studi Kasus Implementasi Graf Berbobot
Sumber: dokumentasi pribadi

Berdasarkan visualisasi graf berbobot pada gambar 15, simpul Akun3, Akun4, dan Akun5 memiliki keterhubungan yang lebih intensif antara satu simpul dengan yang lain, dengan intensitas interaksi bernilai lebih besar dari 10. Ketiga simpul tersebut membentuk *cluster* yang menandakan hubungan lebih dekat dibandingkan dengan simpul lainnya, yaitu simpul Akun1 dan Akun2, dengan intensitas interaksi bernilai lebih kecil dari 4. Algoritma media sosial berbasis interaksi akan mengutamakan

konten kiriman dari pengguna dalam satu *cluster* untuk ditampilkan, sehingga pengguna Akun4 akan lebih sering melihat kiriman pengguna Akun3 dan Akun5, karena ketiga pengguna berada dalam satu *cluster*, dibandingkan kiriman pengguna Akun1 dan Akun2.

2. Operasi *bot farms* dalam memanipulasi algoritma

Untuk memodelkan *bot farms*, dilakukan pembaruan terhadap simulasi dan sisi yang sebelumnya telah dilakukan. Sebagai contoh studi kasus, dibentuk 3 buah akun bot dengan nama akun bot dan interaksi antar akun bot sebagai berikut.

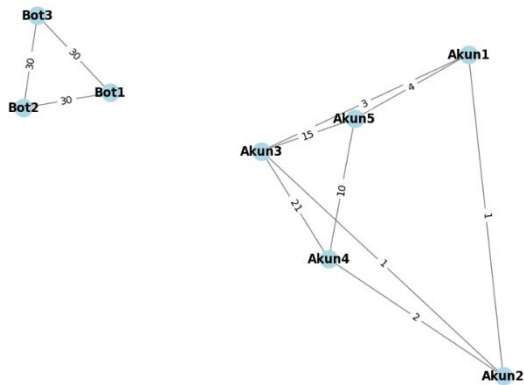
```
# implementasi graf
G = nx.Graph()

# menambahkan akun sebagai simpul
nodes = ['Akun1', 'Akun2', 'Akun3', 'Akun4', 'Akun5', 'Bot1', 'Bot2', 'Bot3']
G.add_nodes_from(nodes)

# menambahkan intensitas interaksi sebagai sisi berbobot
edges = [
    ('Akun1', 'Akun2', 1), ('Akun1', 'Akun3', 3),
    ('Akun1', 'Akun5', 4), ('Akun2', 'Akun3', 1),
    ('Akun2', 'Akun4', 2), ('Akun3', 'Akun4', 21),
    ('Akun3', 'Akun5', 15), ('Akun4', 'Akun5', 10),
    ('Bot1', 'Bot2', 30), ('Bot1', 'Bot3', 30),
    ('Bot2', 'Bot3', 30)
]
G.add_weighted_edges_from(edges)
```

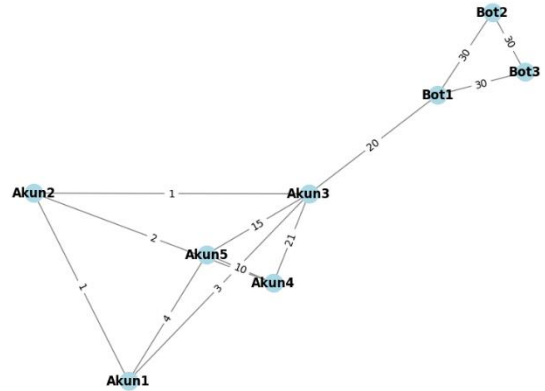
Gambar 16. Penetapan Akun sebagai Simpul dan Interaksi sebagai Sisi Berbobot dengan Bot
Sumber: dokumentasi pribadi

Implementasi *bot farms* merekayasa interaksi yang tinggi antar akun bot untuk membentuk *cluster* berisikan akun-akun bot. Visualisasi graf berbobot dengan eksistensi *bot farms* adalah sebagai berikut.



Gambar 17. Visualisasi Studi Kasus Implementasi Graf Berbobot dengan Bot
Sumber: dokumentasi pribadi

Melalui manipulasi interaksi, *cluster* beranggotakan simpul akun bot memiliki hubungan internal yang lebih kuat dibandingkan *cluster* ketiga simpul akun pengguna sebelumnya. Akibatnya, algoritma media sosial berbasis interaksi menganggap bahwa konten yang disiarkan antara akun bot merupakan konten yang relevan untuk disebarluaskan ke pengguna lainnya. Sangat mungkin bagi akun pengguna untuk menerima kiriman konten tersebut dan berinteraksi dengan akun-akun bot, membentuk intensitas interaksi yang tinggi, khususnya terhadap konten yang disetujui oleh pengguna sehingga terbentuk fenomena *filter bubble* dan *echo chamber*, dan membuka jalan bagi bot untuk menyebarkan hoaks secara luas.



Gambar 18. Visualisasi Studi Kasus Implementasi Graf Berbobot setelah Pengguna Berinteraksi dengan Akun Bot
Sumber: dokumentasi pribadi

Tidak hanya berhenti di pengguna Akun3, berdasarkan intensitas interaksi yang lebih tinggi, algoritma juga akan menampilkan konten yang disiarkan antara akun bot kepada pengguna Akun4 dan Akun5, menyebarkan hoaks dengan lebih luas, membentuk fenomena *filter bubble* dan *echo chamber*, dan menstimulasi polarisasi opini memanfaatkan celah yang dimiliki oleh algoritma media sosial berbasis interaksi.

V. ANALISIS: STRATEGI MITIGASI

Berdasarkan studi kasus yang telah disampaikan, terlihat bahwa dengan memanipulasi algoritma media sosial berbasis interaksi, *bot farms* dapat dengan mudah membentuk hubungan dengan pengguna media sosial, menjadikannya saluran untuk menyebarkan hoaks secara tidak bertanggung jawab. Oleh sebab itu, diperlukan beberapa strategi yang mampu memitigasi dampak dari peran *bot farms* pada media sosial.

1. Deteksi bot melalui analisis graf

Graf berbobot yang memodelkan hubungan antar pengguna media sosial sebagai dasar algoritma berbasis interaksi dapat dianalisis menggunakan metrik seperti *degree centrality*, *clustering coefficient* dan *temporal analysis* untuk mengidentifikasi pola perilaku yang tidak alami, untuk kemudian dicurigai sebagai *bot farms* dan dilakukan tindakan lebih lanjut.

A. Degree Centrality

Degree centrality mengukur derajat dari setiap simpul, yaitu jumlah sisi yang berhubungan langsung oleh sebuah simpul, untuk memberikan analisis ketidakwajaran perilaku pada sebuah *cluster* [6]. Akun bot pada sebuah *bot farms* akan memiliki derajat yang sangat tinggi sebagai hasil dari interaksi otomatis dengan taraf signifikan dalam waktu singkat untuk menciptakan ilusi *cluster* dengan hubungan internal tinggi. Menggunakan *degree centrality*, simpul dengan jumlah sisi yang tidak wajar dapat diidentifikasi sebagai indikasi akun bot, misalnya akun dengan derajat tinggi, tetapi terhubung dengan sebagian besar akun yang tidak aktif atau akun bot lainnya. Ketidakwajaran juga muncul untuk kondisi banyak akun memiliki nilai *degree centrality* yang ekuivalen, mengindikasikan eksistensi *bot farms*

berdasarkan hubungan interaksi yang terbentuk antar setiap akun bot.

Tabel II. Contoh Nilai *Degree Centrality* untuk Graf Berbobot

Simpul	Akun1	Akun2	Akun3	Bot1	Bot2	Bot3	Bot4
<i>Degree Centrality</i>	0.33	0.17	0.17	0.50	0.50	0.50	0.50

Sebagai contoh, pada distribusi nilai *degree centrality* yang diilustrasikan pada tabel II, akun pengguna memiliki nilai *degree centrality* yang beragam, sedangkan akun bot memiliki nilai yang ekuivalen, menandakan indikasi yang tinggi bahwa terdapat eksistensi *bot farms* pada graf berbobot tersebut.

B. Clustering Coefficient

Clustering coefficient mengukur tingkatan hubungan internal pembentuk *cluster* dalam sebuah graf. Nilai *clustering coefficient* yang tinggi menunjukkan bahwa simpul berada dalam sebuah *cluster* dengan banyak hubungan internal. Walaupun tidak selalu indikatif, dalam konteks *bot farms*, bot akan membentuk *cluster* dengan hubungan internal sangat tinggi dan hubungan ke luar *cluster* minimal. Menggunakan perhitungan *clustering coefficient*, dapat dilakukan identifikasi terhadap *cluster* yang mencurigakan dalam graf, yaitu *cluster* dengan hubungan internal padat tetapi hubungan eksternal rendah.

Tabel III. Contoh Nilai *Clustering Coefficient* untuk Graf Berbobot

Simpul	Akun1	Akun2	Akun3	Bot1	Bot2	Bot3	Bot4
<i>Degree Centrality</i>	0.33	0.17	0.17	0.50	0.50	0.50	0.50

Ilustrasi distribusi nilai *clustering coefficient* pada tabel III menunjukkan bahwa akun pengguna memiliki nilai *clustering coefficient* yang beragam sementara akun bot memiliki nilai yang cenderung tidak lazim. Nilai ini mencerminkan *cluster bot farms* yang terbentuk, dimana setiap bot dalam *bot farms* hanya berhubungan satu dengan yang lain, tanpa memiliki hubungan eksternal dengan simpul lain.

Kedua strategi mitigasi sebelumnya dapat diimplementasikan menggunakan library NetworkX yang memiliki fungsi *degree centrality* dan *clustering* untuk menghasilkan nilai-nilai analisis graf berbobot.

```

degree_centrality = nx.degree_centrality(G)
print("Degree Centrality:")
for node, centrality in degree_centrality.items():
    print(f"{node}: {centrality}")

clustering_coeff = nx.clustering(G)
print("\n\nClustering Coefficient:")
for node, coeff in clustering_coeff.items():
    print(f"{node}: {coeff}")

```

Gambar 19. Implementasi Analisis *Degree Centrality* dan *Clustering Coefficient* Menggunakan Library NetworkX
Sumber: dokumentasi pribadi

Namun, karena studi kasus hanya menggunakan 5 akun pengguna dan 3 akun bot dengan hubungan sederhana, hasil analisis yang disampaikan kurang signifikan untuk memberikan indikasi akun bot yang akurat. Hasil analisis mampu memberikan indikasi yang semakin akurat untuk graf yang lebih besar dan kompleks.

```

Degree Centrality:
Akun1: 0.42857142857142855
Akun2: 0.42857142857142855
Akun3: 0.7142857142857142
Akun4: 0.42857142857142855
Akun5: 0.42857142857142855
Bot1: 0.42857142857142855
Bot2: 0.2857142857142857
Bot3: 0.2857142857142857

Clustering Coefficient:
Akun1: 0.6666666666666666
Akun2: 0.6666666666666666
Akun3: 0.4
Akun4: 0.6666666666666666
Akun5: 0.6666666666666666
Bot1: 0.3333333333333333
Bot2: 1.0
Bot3: 1.0

```

Gambar 20. Hasil Analisis *Degree Centrality* dan *Clustering Coefficient* untuk Graf Berbobot Studi Kasus
Sumber: dokumentasi pribadi

C. Temporal Analysis

Temporal analysis melakukan analisis terhadap pola waktu interaksi yang terjadi antar pengguna untuk mengidentifikasi aktivitas interaksi yang tidak alami. *Bot farms* umumnya menunjukkan pola temporal yang sinkron dan seragam, seperti banyak interaksi dilakukan berulang-ulang dengan interval waktu tetap. Pemetaan waktu interaksi pada setiap sisi yang dimiliki graf akan dapat memperlihatkan pola temporal yang tidak wajar, mengindikasikan eksistensi *bot farms*.

2. Diversifikasi konten menggunakan sistem *epsilon-greedy*

Bot farms mengeksploitasi algoritma media sosial berbasis interaksi untuk melakukan tindakan penyebaran hoaks. Maka, strategi dapat dilakukan terhadap algoritma untuk memutakhir algoritma agar konten yang ditampilkan kepada pengguna tidak sepenuhnya berdasarkan interaksi saja, menghambat terbentuknya fenomena *filter bubble* dan *echo chamber* yang berdampak fatal dalam konteks penyebaran hoaks. *Epsilon-greedy* sebagai strategi dalam pengambilan keputusan berbasis pengaturan eksplorasi-eksploitasi dapat digunakan untuk melakukan diversifikasi konten pada algoritma media sosial.

Strategi *epsilon-greedy* bekerja dengan memperkenalkan elemen eksplorasi ke dalam algoritma media sosial:

- Dengan probabilitas $1 - \epsilon$, algoritma akan menggunakan basis interaksi untuk menampilkan konten kepada pengguna;
- Dengan probabilitas ϵ , algoritma akan menampilkan konten yang beragam dan tidak sepenuhnya didasarkan terhadap interaksi yang dimiliki pengguna.

Dengan menyeimbangkan eksplorasi dan eksploitasi, *epsilon-greedy* membantu memitigasi risiko terbentuknya fenomena *filter bubble* dan *echo chamber* dengan menampilkan informasi yang lebih beragam dan memperluas pandangan pengguna.


```

import random

# simulasi menampilkan konten berdasarkan interaksi atau konten beragam
konten = [
    {"id": 1, "kategori": "basis interaksi"},
    {"id": 2, "kategori": "konten beragam"}
]

# algoritma didasarkan terhadap basis interaksi
preferensi_pengguna = "basis interaksi"

# parameter epsilon untuk probabilitas eksplorasi
epsilon = 0.35

def strategi_epsilon_greedy(konten, preferensi, epsilon):

    # eksplorasi
    if random.random() < epsilon:
        eksplorasi = [k for k in konten if k["kategori"] != preferensi]
        if eksplorasi:
            return random.choice(eksplorasi)

    # eksploitasi
    eksplorasi = [k for k in konten if k["kategori"] == preferensi]
    if eksplorasi:
        return random.choice(eksplorasi)

    # eksplorasi penuh untuk kondisi nilai random luput
    return random.choice(konten)

# simulasi algoritma
algoritma = [strategi_epsilon_greedy(konten, preferensi_pengguna, epsilon) for _ in range(10)]

for i, r in enumerate(algoritma, 1):
    print(f"Konten ke-{i} yang ditampilkan algoritma: {r['kategori']}")

```

Gambar 21. Simulasi Strategi Sistem *Epsilon-greedy*
 Sumber: dokumentasi pribadi

```

Konten ke-1 yang ditampilkan algoritma: basis interaksi
Konten ke-2 yang ditampilkan algoritma: basis interaksi
Konten ke-3 yang ditampilkan algoritma: basis interaksi
Konten ke-4 yang ditampilkan algoritma: basis interaksi
Konten ke-5 yang ditampilkan algoritma: basis interaksi
Konten ke-6 yang ditampilkan algoritma: konten beragam
Konten ke-7 yang ditampilkan algoritma: basis interaksi
Konten ke-8 yang ditampilkan algoritma: basis interaksi
Konten ke-9 yang ditampilkan algoritma: konten beragam
Konten ke-10 yang ditampilkan algoritma: basis interaksi

```

Gambar 22. Hasil Simulasi Strategi Sistem *Epsilon-greedy*
 Sumber: dokumentasi pribadi

3. Akses akun terpercaya

Algoritma media sosial berbasis interaksi dapat dimodifikasi lebih lanjut untuk memberikan bobot lebih tinggi pada akun resmi terpercaya untuk meningkatkan integritas informasi yang disebar pada *platform* media sosial. Akun resmi seperti organisasi berita terverifikasi, lembaga pemerintah, atau tokoh masyarakat dapat diberikan atribut kepercayaan yang secara eksplisit meningkatkan prioritas distribusi konten yang dikirimkan akun-akun tersebut. Dalam pemodelan graf berbobot, simpul yang merepresentasikan akun terpercaya dapat diberikan bobot awal yang lebih tinggi kepada pengguna lainnya, sehingga interaksi yang melibatkan akun-akun tersebut akan menghasilkan sisi berbobot tinggi. Upaya ini mengarahkan algoritma untuk mendistribusikan konten dari akun terpercaya dibandingkan konten yang dimanipulasi oleh *bot farms*, meningkatkan integritas konten yang terdistribusi pada media sosial.

4. Pendidikan digital

Selain meletakkan fokus kepada inovasi yang dapat dilakukan kepada algoritma, pengguna juga perlu untuk menjadi ‘pribumi digital’ yang cerdas dan kritis dalam berselancar di media sosial. Pendidikan digital dapat membantu pengguna untuk mengembangkan kemampuan kritis dalam menganalisis dan mengevaluasi informasi, bahkan dari sumber yang dikenali untuk memiliki pandangan yang serupa. Fenomena *filter bubble* dan *echo chamber* yang menyebabkan bias konfirmasi dan polarisasi opini dapat ditangani dengan mengajarkan pengguna untuk mengenali tanda-tanda hoaks yang merugikan.

Pendidikan digital juga menekankan pentingnya pemeriksaan fakta terhadap informasi yang didengar.

Kemampuan literasi digital menjadi urgensi krusial untuk dimiliki di era digital. Dengan kemampuan ini, pengguna dapat lebih waspada terhadap informasi dari konten yang ditampilkan algoritma media sosial, khususnya terhadap akun yang dipercaya sebagai akibat dari selarasnya pandangan yang dimiliki. Hal ini diharapkan mampu membangun masyarakat digital yang lebih kritis dan tangguh menghadapi penyebaran hoaks di media sosial.

VI. KESIMPULAN

Media sosial modern yang didasarkan terhadap algoritma berbasis interaksi telah membawa perubahan signifikan terhadap cara individu dan masyarakat berinteraksi, berkomunikasi, dan menyebarkan informasi. Namun, implementasi algoritma ini turut menghadirkan tantangan serius, yaitu penyebaran hoaks yang diperkuat oleh fenomena *filter bubble* dan *echo chamber*. Eksploitasi terhadap jaringan terorganisir akun-akun bot sebagai *bot farms* memanipulasi pola interaksi dan membentuk pseudo-komunitas menunjukkan bagaimana algoritma media sosial berbasis interaksi dapat disalahgunakan untuk menyebarkan disinformasi secara luas, meningkatkan risiko polarisasi opini, dan mengancam stabilitas sosial masyarakat digital.

Menggunakan graf berbobot sebagai pendekatan pemodelan, makalah ini telah menunjukkan kerentanan algoritma media sosial terhadap manipulasi *bot farms*. Hasil analisis meningkatkan urgensi untuk mengembangkan strategi mitigasi yang efektif, meliputi deteksi pola graf untuk mengidentifikasi aktivitas bot memanfaatkan analisis *degree centrality*, *clustering coefficient*, dan *temporal analysis*, diversifikasi konten menggunakan pendekatan strategi *epsilon-greedy*, transparansi algoritma untuk memprioritaskan akun terpercaya, serta pendidikan digital untuk meningkatkan literasi dan kesadaran pengguna terhadap hoaks.

Penerapan strategi-strategi tersebut secara konsisten diharapkan mampu menjaga integritas ruang digital dengan meminimalisir dampak *bot farms* dan memastikan bahwa media sosial mendukung distribusi informasi yang kredibel dan jujur. Dengan menempatkan kebenaran informasi sebagai pilar utama, ekosistem digital dapat berkontribusi positif terhadap perkembangan masyarakat digital yang lebih kritis, inklusif, dan tangguh terhadap disinformasi.

VII. UCAPAN TERIMA KASIH

Penulis memanjatkan puji syukur kepada Tuhan yang Maha Esa atas berkat dan rahmat-Nya yang melimpah dalam proses penulisan makalah sehingga makalah dapat diselesaikan dengan baik dan tepat waktu. Penulis juga mengucapkan terima kasih yang sebesar-besarnya kepada Bapak Dr. Ir. Rinaldi, M. T. selaku dosen pengampu mata kuliah IF1220 Matematika Diskrit kelas K01 atas waktu, bimbingan, dan ilmu pengetahuan yang telah diberikan sebagai bekal dalam pembentukan makalah. Akhir kata, penulis menyampaikan dukungan semangat kepada teman-teman penulis yang sedang berjuang bersama-sama untuk menyelesaikan masa studi semester ganjil tahun ajaran 2024/2025.

VIII. LAMPIRAN

Source code program yang dibentuk sebagai pendukung bagian studi kasus dan analisis dapat diakses melalui tautan berikut:

<https://github.com/rafenmaxxx/StudiKasusGrafBerbobot.git>

REFERENCES

- [1] Munir, Rinaldi. "Homepage Rinaldi Munir". <https://informatika.stei.itb.ac.id/~rinaldi.munir/>. (diakses pada 4 Januari 2024)
- [2] yWorks. (n.d.). Clustering graphs and networks. yWorks. <https://www.yworks.com/pages/clustering-graphs-and-networks>.
- [3] Lin, X., Shang, T., & Liu, J. (2014). An estimation method for relationship strength in weighted social network graphs. *Journal of Computer and Communications*, 2(04), 82.
- [4] Liu, R., Feng, S., Shi, R., & Guo, W. (2014). Weighted graph clustering for community detection of large social networks. *Procedia Computer Science*, 31, 85-94.
- [5] Haiz (2024, September 09). *The Rise of Phone Bot Farms: A Shortcut to Social Media Fame*. Wake Up Singapore. <https://wakeup.sg/phone-bot-farms-social-media-fame/>.
- [6] Zhang, J., & Luo, Y. (2017, March). Degree centrality, betweenness centrality, and closeness centrality in social network. In *2017 2nd international conference on modelling, simulation and applied mathematics (MSAM2017)* (pp. 300-303). Atlantis press.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 7 Januari 2025



Rafen Max Alessandro
13523031