

# Aplikasi Graph pada Graph Neural Network

Zaidan Naufal Sudrajat 13518021  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13518021@std.stei.itb.ac.id

**Abstract**—Machine Learning merupakan salah satu teknologi yang sangat berkembang pada saat ini. Terdapat dua jenis machine learning yaitu unsupervised machine learning, dan supervised machine learning. Supervised machine learning merupakan teknik machine learning saat input yang dimasukkan dilabeli oleh pemogram. Salahsatu dari algoritma supervised machine learning adalah Graph Neural Network. Pada makalah ini akan dijelaskan mengenai introduksi Graph Neural Network beserta aplikasi dari graph pada Graph Neural Network

**Keywords**—Graph Neural Network, Machine Learning , Graph

## I. PENDAHULUAN

Machine Learning merupakan salah satu teknologi pada bidang computer science yang memungkinkan sebuah algoritma program dapat berkembang melalui pengalaman dan latihan [1]. Generasi awal machine learning sudah ada sejak abad 20. Salah satunya adalah Perceptron yang dibuat pada tahun 1958 dibuat oleh Frank Rosnblatt. Perceptron dibuat dengan tujuan program dapat menemukan pola dan menemukan bentuk [1].

Sekian majunya waktu semakin banyak pengaplikasian machine learning. Pada saat makalah ini dibuat Machine Learning dan AI merupakan buzzword yang sangat sering digaungkan. Banyak perusahaan multinasional memanfaatkan teknologi ini untuk keperluan bisnisnya dan produknya. Salah satu contohnya yaitu fitur voice recognition yang dimiliki oleh AI Siri milik Apple dan fitur OKGoogle pada Handphone Android.

Dengan ada nya *voice recognition* suatu perangkat dapat mengenali kata kata yang diucapkan oleh pengguna , yang akan dilanjutkan dengan mengartikan kata-kata itu sebagai sebuah perintah baik baik perintah mengaktifkan alarm atau mencari suatu item di internet. Jika kita perhatikan baik baik , membuat sebuah algoritma yang dapat mendeteksi suara dan menjadikannya sebuah string yang akan diproses saja merupakan sebuah algoritma yang cukup sulit untuk diimplementasikan pada pemrograman konvensional. Untuk mengetahui suatu kata dibutuhkan banyak sekali parameter yang diperlukan seperti kecepatan bunyi, panjang gelombang suara, dan lainnya. Belum juga terdapat masalah noise yang menghalangi suara dan juga aksentuasi setiap orang dan tinggi rendahnya nada dari seseorang akan makin memperbanyak parameter yang dibutuhkan untuk melakukan voice recognition pada algoritma konvensional.

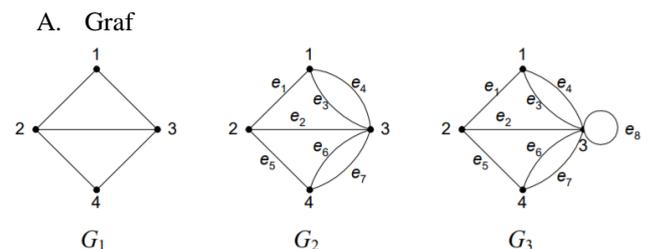
Sudah cukup terlihat masalah yang harus dihadapi pemogram untuk mendeteksi kata, misalkan kita berhasil melakukan hal

tersebut untuk sebuah kata. Kabar buruknya terdapat minimal 12700 kata yang terdapat di Bahasa Indonesia, Bagaimana dengan Bahasa lainnya seperti Bahasa Inggris ?. Butuh berapa banyak programmer dan waktu yang dibutuhkan untuk mengimplementasi hal tersebut? Sepertinya tidak masuk akal untuk dilakukan.

Untuk itulah adanya Machine Learning , dengan algoritma Machine learning , parameter akan di “generate” oleh algoritma itu sendiri berupa model Neural Network, sehingga hal seperti *voice recognition* dapat diimplementasikan. Contoh lain Machine Learning adalah RTX Voice dari Nvidia yang memungkinkan menghapus *background voice* dan *noise* dari video call secara langsung pada saat dijalankan, Serta pengambilan data KTP secara otomatis dari kamera Handphone pada aplikasi seperti Gojek dan OVO.

Pada Makalah ini akan diberikan suatu contoh model machine learning Neural Network yang memanfaatkan data struktur graf.

## II. TEORI DASAR



Gambar 1 : Contoh Graf  
dikutip dari

<http://informatika.stei.itb.ac.id/~rinaldi.munir/> [4]

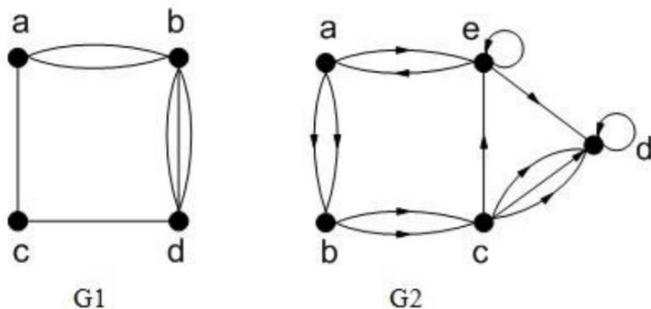
Graf merupakan cara untuk merepresentasikan objek-objek diskrit dan hubungan antara objek-objek tersebut[4]. Suatu graf ( $G$ ) terdiri dari dua bagian yaitu simpul ( $V$  (Vertices)) dan sisi ( $E$  (Edge)), Graf dapat di tuliskan dengan :

$$G = (V, E)$$

dimana  $V$  adalah himpunan tidak kosong dari vertices dan  $E$  adalah edge yang menghubungkan simpul.

Graf dapat dikelompokkan menjadi beberapa jenis. Berdasarkan adanya gelang atau sisi ganda pada graf, graf dibagi menjadi dua jenis yaitu graf sederhana atau graf yang tidak memiliki sisi ganda, dan graf sisi ganda yang merupakan kebalikannya yaitu graf yang memiliki simpul ganda, terdapat jenis khusus pada graf ganda yaitu graf semu atau graf yang memiliki sisi gelang. Berdasarkan orientasi arah graf dapat

terbagi menjadi graf berarah dan graf tidak berarah. Berikut merupakan contoh gambar dari graf berarah dan tidak berarah.



bahwa simpul 1 bertetangga dengan 2 dan 3 , simpul 2 bertetangga dengan 1 dan 3, simpul 3 bertetangga dengan 1,2 dan 4 , simpul 4 hanya bertetangga dengan simpul 3 , dan simpul 5 tidak bertetangga dengan simpul manapun.

G1 : graf tak-berarah; G2 : Graf berarah

Gambar 2 Graf Berarah dan Tidak Berarah dikutip dari

<http://informatika.stei.itb.ac.id/~rinaldi.munir/> [4]

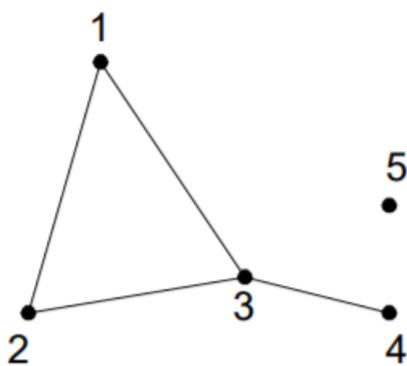
Berikut merupakan table untuk memudahkan pengkelompokan jenis graf

| Jenis              | Sisi        | Sisi ganda dibolehkan? | Sisi gelang dibolehkan? |
|--------------------|-------------|------------------------|-------------------------|
| 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                      |

Tabel Pengelompokan Graf

dikutip dari <http://informatika.stei.itb.ac.id/~rinaldi.munir/> [4]

### B. Terminologi dari graf



Gambar 3 Contoh Graf dikutip dari

<http://informatika.stei.itb.ac.id/~rinaldi.munir/> [4]

Gambar di atas akan digunakan sebagai penggambaran dari terminology graf. Berikut merupakan beberapa terminologi yang relevan dengan lingkup makalah ini.

#### 1. Ketetangaan (*adjacent*)

Dua buah simpul bertetangga apabila dua simpul tersebut terhubung langsung. Berdasarkan gambar dapat diketahui

#### 2. Bersisian (*incidency*)

Sembarang sisi  $e = (v_j, v_k)$  dikatakan e bersisian dengan simpul  $v_j$  dan e bersisian dengan simpul  $v_k$ . Apabila digunakan graf sebelumnya dapat diketahui bahwa sisi (1,2) bersisian dengan sisi 1 dan sisi (1,2) bersisian dengan sisi 2, akan tetapi sisi (3,4) tidak bersisian dengan sisi 1

#### 3. Simpul terpercil

Simpul terpercil merupakan simpul yang tidak terhubung dengan simpul manapun atau dalam kata lain tidak ada sisi yang bersisian dengan simpul tersebut. Simpul 5 merupakan contoh simpul terpercil

#### 4. Graf Kosong (*Null Graph*)

Graf Kosong adalah graf yang memiliki himpunan kosong sebagai sisi. Berikut contoh graf kosong.



Gambar 4 Graf Kosong dikutip dari

<http://informatika.stei.itb.ac.id/~rinaldi.munir/> [4]

#### 5. Derajat

Derajat sebuah Simpul Graf adalah jumlah sisi yang bersisian dengan simpul tersebut. Pada gambar simpul 1 dan 2 memiliki derajat 2, simpul 3 memiliki derajat 3 simpul 4 memiliki derajat 1 , dan simpul 5 memiliki derajat 0.

### C. Representasi Graf

Agar lebih mudah dihitung dan dikelola pada computer graf dapat direpresentasikan menjadi berbentuk matriks. Berikut merupakan beberapa matriks representasi graf.

#### 1. Matriks ketetangaan (*adjacency matriks*)

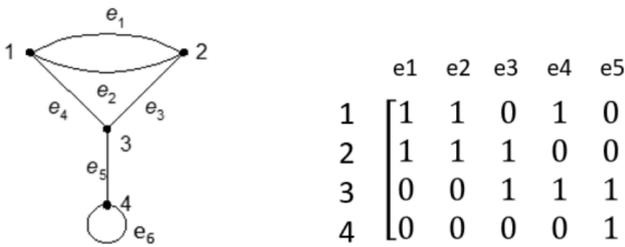
Misalkan A adalah representasi matriks dari graph maka  $a_{ij}$  akan bernilai 1 apabila simpul i dan j bertetangga dan bernilai 0 apabila simpul i dan j tidak bertetangga.

|   |   |   |   |   |   |
|---|---|---|---|---|---|
|   | 1 | 2 | 3 | 4 | 5 |
| 1 | 0 | 1 | 1 | 0 | 0 |
| 2 | 1 | 0 | 1 | 0 | 0 |
| 3 | 1 | 1 | 0 | 1 | 0 |
| 4 | 0 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 |

Representasi Matriks Ketetanggan Graf Gambar 3  
dikutip dari <http://informatika.stei.itb.ac.id/~rinaldi.munir/> [4]

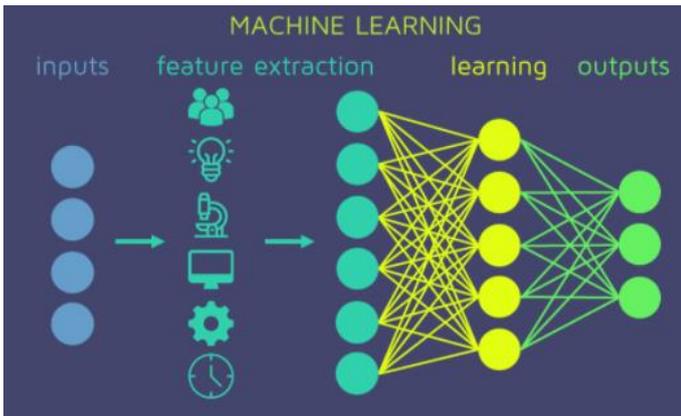
2. Matriks Bersisian (*incidency matrix*)

Misalkan A adalah sebuah matriks representasi graf maka  $a_{ij}$  akan bernilai 1 apabila simpul i bersisian dengan j dan bernilai 0 apabila simpul I tidak bersisian dengan simpul j.



Gambar 5 Graf dan Representasi Matriks Bersisiannya  
dikutip dari  
<http://informatika.stei.itb.ac.id/~rinaldi.munir/>

D. Machine Learning



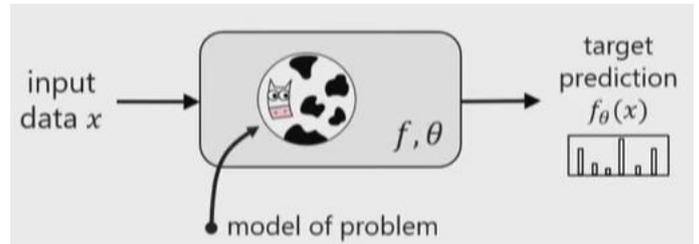
Gambar 6 Penggambaran Machine Learning  
dikutip dari :  
<https://quantdare.com/>

Machine Learning merupakan salah satu bagian dari teknologi artificial intelligence (AI) yang memungkinkan suatu algoritma berkembang menjadi lebih baik dari latihan dan pengalaman tanpa di program secara eksplisit. [1].

Terdapat dua jenis Machine Learning yaitu Unsupervised Learning dan Supervised Learning. Pada Unsupervised Learning input data yang dimasukkan kedalam model tidak dilabeli terlebih dahulu, jenis ini pada umumnya digunakan

dengan tujuan mencari pola tersembunyi atau pola abstrak yang dimiliki oleh data dan sulit (memakan waktu lama) apabila dilakukan oleh manusia. Sedangkan Supervised Learning input data awal yang dilabeli. Graph Neural Network merupakan salah satu dari jenis Supervised Learning , sehingga jenis ini akan dijelaskan lebih lanjut.

Deep Learning merupakan salah satu bagian dari Machine Learning. Deep learning menggunakan Neural Network dengan layer yang banyak.. Neural Network merupakan metode pemodelan data dengan mengikuti bentuk neuron pada otak manusia. Neuron – Neuron ini saling berhubung satu sama lain.



Gambar 6 Penggambaran Supervised Learning  
dikutip dari  
[https://www.microsoft.com/en-us/research/\[5\]](https://www.microsoft.com/en-us/research/[5])

Sebagai contoh Deep Learning pada Supervised Learning adalah seperti bagaimana kita mengajari seorang anak kecil untuk mengenali setiap jenis mainan, kita memberi anak kecil 100 mainan terdiri atas 10 jenis mobil mainan, 10 jenis boneka beruang, 10 jenis motor mainan dan jenis mainan lainnya. Lalu kita meminta anak kecil tersebut untuk memperhatikan karakteristik dari tiap jenis mainan, seperti mobil memiliki 4 roda, atau boneka beruang yang memiliki mata. Setelah itu kita memberi mainan yang bukan bagian dari 100 mainan yang diberikan sebelumnya , dan meminta anak tersebut untuk menebak termasuk jenis apa mainan ini, lakukan hal tersebut terus menerus hingga anak dapat menebak dengan tingkat kebenaran yang tinggi.

Bagaimana dengan komputer , seperti pada kasus anak kecil sebelumnya pertama diperlukan adanya dataset inout. Dimisalkan terdapat input dataset berupa matriks vector  $\{(x_1,y_1), \dots, (x_n,y_n)\}$ . Setiap input dataset tersebut telah dilabeli sesuai klasifikasinya masing masing . data tersebut diinput ke algoritma untuk dipelajari oleh computer dan disimpan sebagai sebuah model , dataset input untuk pelatihan disebut dataset training. Lalu seperti sebelumnya model tersebut diuji dengan dataset yang baru, diulang terus menerus hingga mendapatkan tingkat ketepatan yang tinggi, dataset yang diberikan untuk pengujian dinamai dataset uji.

Bagaimana caranya tingkat ketepatan sebuah model makin bertambah dari setiap iterasi pembelajaran dan ujinya? Digunakanlah sebuah Loss Function. Loss Function menggambarkan tingkat akurasi dari sebuah model semakin kecil loss maka semakin akurat sebuah model.

$$\mathcal{L}(\theta) = \frac{1}{N} \sum_i L(f_{\theta}(x_i), y_i)$$

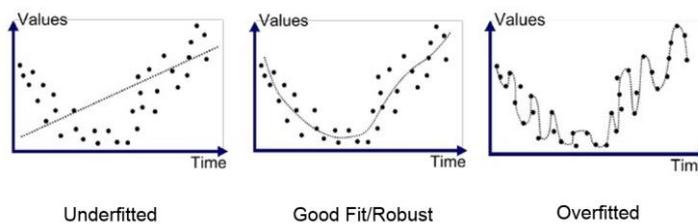
Gambar 7 Loss Function

dikutip dari

[https://www.microsoft.com/en-us/research/\[5\]](https://www.microsoft.com/en-us/research/[5])

Dilakukannya pembelajaran secara berulang adalah untuk mencari sebuah  $\theta$  yang sebagaimana sehingga dapat memunculkan sebuah loss yang minimum. Loss minimum dapat dicari menggunakan Gradient Descent pada model.

Model yang diinginkan pada akhirnya adalah model yang tidak *underfitting* atau model yang tidak berhubungan dengan dataset (biasanya tingkat loss nya masih tinggi), atau pun *overfitting* yaitu model yang terlalu mirip dengan dataset learning dan dataset uji sehingga jika diberi data tes diluar yang dataset training dan dataset uji akan mengeluarkan akurasi rendah.



Gambar 8 Representasi Model pada Dataset

Dikutip dari

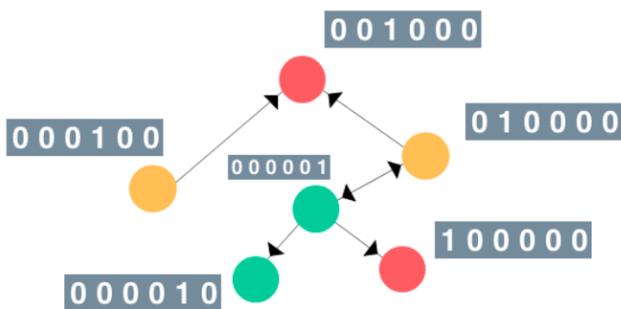
<https://medium.com/greyatom/>

### III. GRAPH NEURAL NETWORK

#### A. Sejarah Singkat Graph Neural Network

Graph Neural Network pertama kalinya dikemukakan oleh Scarselli dan tim, pada paper nya (Scarselli, F., Gori, M., Tsoi, A., Hagenbuchner, M. & Monfardini, G. 2009, 'The graph neural network model') di tahun 2009.

Graph Neural Network merupakan sebuah tipe dari Neural Network yang langsung menggunakan data struktur graf pada dataset dan outputnya. Umumnya GNN. bertujuan untuk mengklasifikasi setiap simpul pada graf dan makna simpul pada graf tersebut. Karena merupakan tipe supervised learning maka pada dataset training setiap simpul diberi label terlebih dahulu. Lalu model akan dilatih untuk dapat memprediksi label dari setiap simpul.



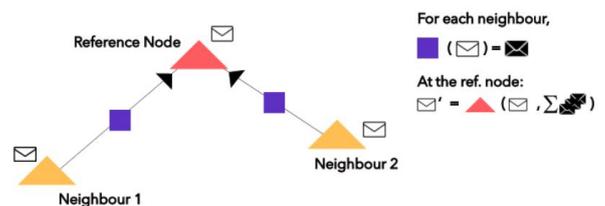
Gambar 9 Contoh Graf yang dilabeli setiap simpulnya

Dikutip dari

<https://medium.com/dair-ai>

#### B. Langkah Kerja Graph Neural Network

Graph Neural Network melakukan training dengan melakukan Neural Message Passing. Misalkan ada tiga simpul A, B, dan C. Simpul A bersisian dengan C, tetapi tidak dengan B, begitupun Simpul B bersisian dengan C, tetapi tidak dengan A. Dimisalkan pula bahwa simpul C adalah Reference Node. Simpul A dan B memproses data (label atau karakteristik) yang dimiliki oleh nya dan mengirimkan data tersebut kesetiap Simpul yang bertetangga dengan Simpul A dan B. Karena pada contoh ini A dan B hanya bertetangga dengan C maka pesan berupa data hanya akan dikirimkan ke C (pada contoh ini C dijadikan reference node dan dianggap tidak mengirimkan pesan data). Simpul C akan menerima setiap message yang dikirim lalu memprosesnya dan mengupdate data pada C dengan data baru hasil proses dari semua message dan data C sebelumnya. Begitulah cara kerja Neural Message Passing (NMP).



Gambar 10 Penggambaran Neural Message Passing

Dikutip dari

<https://medium.com/dair-ai>

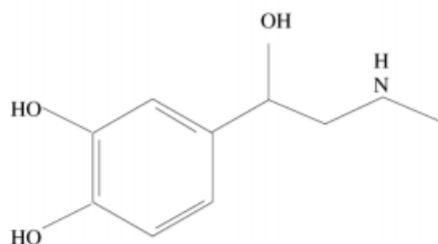
Pada satu waktu NMP akan dilakukan setiap simpul ke semua tetangganya secara paralel. NMP ini akan dilakukan terus menerus sehingga data pada simpul akan terus terupdate dengan data dari tetangganya. Jumlah waktu yang diperlukan untuk melakukan NMP merupakan bagian dari perhitungan yang kasus dependent sehingga merupakan bukan lingkup dari makalah ini. Bagaimana cara melakukan pemrosesan data juga merupakan bagian yang sangat kasus dependent biasanya dapat dilakukan dengan menggunakan loss function seperti binary cross entropy, dll Pemilihan loss function juga bukan lingkup dari makalah ini.

Setelah dilakukan cukup training setiap simpul akan mempelajari lebih baik tentang simpulnya sendiri dan simpul tetangganya. Setelah dilakukan N-training setiap simpul akan mempunyai data dan mengenali setiap data dari semua simpul pada graf. Terakhir dapat dilakukan penjumlahan dari semua data pada setiap node sehingga dapat diketahui representasi graf.

#### Kesimpulan Langkah Kerja Graph Neural Network

1. Jadikan setiap simpul sebagai unit yang dapat diupdate datanya dan setiap sisi sebagai lintasan pengiriman message
2. Melakukan N-buah Neural Message Passing untuk setiap simpul
3. Menjumlahkan data dari setiap simpul untuk mendapatkan representasi graf.

### C. Kegunaan Graph Neural Network dalam Machine Learning

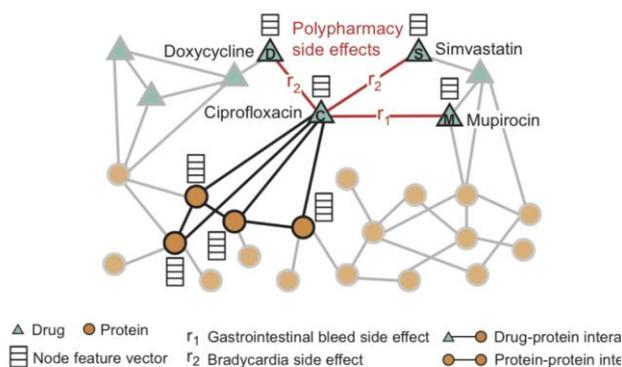


Gambar 10 Senyawa Kimia Adrenaline  
Dikutip dari

Scarselli, F., Gori, M., Tsoi, A., Hagenbuchner, M. & Monfardini, G. 2009, 'The graph neural network model'

Pada Paper awal Graph Neural Network, GNN diusulkan digunakan untuk machine learning dalam meniru kasus data yang bentuk graf seperti pada senyawa kimia, atau hubungan website. Pada saat ini terdapat beberapa pengaplikasian Graph Neural Network yaitu

#### 1. Memprediksi Efek samping dari interaksi obat



Gambar 11 Contoh input graf dari Penelitian dari Stanford  
Dikutip dari  
<https://blog.fastforwardlabs.com/>

Pada penelitian ini tim dari Stanford membuat model machine learning mengenai efek samping dari obat (transcribed drug) yang diminum oleh lansia. Penelitian ini dilaksanakan karena tidak dimungkinkan untuk menguji interaksi antar setiap obat satu sama lainnya.

Pada awalnya dilakukan klasifikasi pada model machine learning yang lain akan tetapi mendapat hasil yang terbatas. Hasil tersebut dikarenakan hasil yang dihasilkan adalah lingkupan dari interaksi obat tanpa mengkarakteristikan inti dari interaksi obat tersebut, sehingga model tersebut hanya dapat memodelkan interaksi dari dua obat.

Dengan menggunakan GNN tim dari Stanford dapat memodelkan sebuah interaksi lebih dari dua obat, model ini memiliki performa yang lebih baik dari model lainnya dan mampu mengidentifikasi efek samping yang tidak terjadi apabila hanya meminum satu buah obat saja.

#### 2. Memprediksi model dengan data non-euclidean

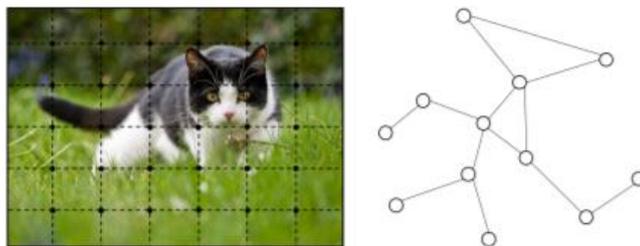


Fig. 1. Left: image in Euclidean space. Right: graph in non-Euclidean space

Gambar 11 Gambar Euclidean dan graf non Euclidean  
Dikutip Dari  
Graph Neural Networks: A Review of Methods and Applications : Jie Zhou and other.

CNN merupakan teknik pemodelan deep learning yang sering dipakai akan tetapi kelemahan dari CNN adalah hanya bisa bekerja pada 2D data dan 1D data seperti gambar dan text. dikarenakan Graph dapat merepresentasikan non-Euclidean, maka teknik ini sangat cocok apabila digunakan untuk membuat pembuatan model machine learning dengan model non-Euclidean

### IV. KESIMPULAN

Machine Learning merupakan sebuah teknologi yang akan terus berkembang. Dengan adanya Graph Neural Network diharapkan menjadi solusi (atau sebuah pilihan solusi) dari pembuatan model machine learning yang dibuat apabila tidak dapat dilakukan dengan baik pada metode neural network lainnya seperti yang terjadi pada kasus data non-euclidean dan kasus kasus lainnya yang sudah dipaparkan sebelumnya.

### V. UCAPAN TERIMAKASIH

Pertama-tama, penulis mengucapkan terimakasih kepada puji syukur kepada Tuhan Yang Maha Esa atas berkat dan rahmatnya, penulis bisa menyelesaikan tugas makalah ini. Penulis juga mengucapkan terimakasih kepada kedua orangtua penulis serta adik dan teman teman Jurusan Informatika yang selalu mendoakan akan kesuksesan dan kelancaran penulis dan turut mendukung penulis dalam menulis makalah ini. Penulis juga mengucapkan terimakasih kepada. Bapak Dr. Ir. Rinaldi Munir, M.T., selaku dosen pengampu mata kuliah IF2120 Matematika Diskrit di kelas 01 yang telah mengajarkan ilmu tentang matematika diskrit, dan juga website beliau yang penulis jadikan bahan acuan dari makalah ini. Penulis pun berterimakasih atas semua pihak yang tidak dapat disebutkan satu per satu yang membantu penulis dalam penyelesaian tugas makalah ini.

### REFERENSI

- [1] <https://www.expert.ai/blog/machine-learning-definition/>
- [2] <https://www.doc.ic.ac.uk/~jce317/history-machine-learning.html>
- [3] <https://www.republika.co.id/berita/koran/didaktika/16/05/09/o6wbc76-kosakata-bahasa-indonesia-hanya-100-ribuE>
- [4] <http://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2020-2021/Graf-2020-Bagian1.pdf>

- [5] <https://www.microsoft.com/en-us/research/video/msr-cambridge-lecture-series-an-introduction-to-graph-neural-networks-models-and-applications/>  
[6] Graph Neural Networks: A Review of Methods and Applications : Jie Zhou and other.  
[7] <https://towardsdatascience.com/a-gentle-introduction-to-graph-neural-network-basics-deepwalk-and-graphsage-db5d540d50b3>  
[8] <https://medium.com/dair-ai/an-illustrated-guide-to-graph-neural-networks-d5564a551783>  
[9] <https://medium.com/towards-artificial-intelligence/>  
[10] <https://blog.fastforwardlabs.com/2019/10/30/exciting-applications-of-graph-neural-networks.html>

### 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, 11 Desember 2020



Zaidan Naufal Sudrajat  
13518021