

# Aplikasi Graf pada *Artificial Neural Network* dan *Backpropagation Algorithm*

Joshua Salimin 13514001  
Program Studi Teknik Informatika  
Sekolah Teknik Elektro dan Informatika  
Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia  
13514001@std.stei.itb.ac.id

**Abstrak**—Makalah ini akan membahas salah satu aplikasi graf yang digunakan pada *artificial neural network*. *Artificial neural network* adalah jaringan (*network*) dari sekelompok unit pemroses kecil (*neural/neuron*) yang dimodelkan dan ditiru berdasarkan seperti jaringan saraf manusia yaitu *neuron*. *Artificial neural network* merupakan suatu sistem yang dapat beradaptasi dengan mengubah struktur-strukturnya dan akan menyelesaikan suatu masalah berdasarkan informasi-informasi yang mengalir baik itu eksternal maupun internal. Aplikasi graf pada *Artificial neural network* terlihat dengan *neuron* input, *neuron output*, dan *neuron* atau unit pemroses sebagai simpul, serta jaringan atau koneksi sebagai sisi-sisinya. Hal ini menggambarkan seperti layaknya jaringan saraf manusia yang terhubung. Makalah ini akan membahas lebih lanjut mengenai aplikasi graf di bidang *artificial neural network* dan algoritma *backpropagation* yang digunakan pada *artificial neural network*.

**Kata Kunci**—*Artificial Neural Network*, *Backpropagation Algorithm*, Graf, dan Kecerdasan Buatan (*Artificial Intelligence*).

## I. LATAR BELAKANG

Kecerdasan buatan (*artificial intelligence*) dapat membuat sebuah komputer seolah-olah “berpikir” layaknya manusia biasa. Dengan demikian, kecerdasan buatan dapat beradaptasi dan menirukan proses belajar manusia yang menyerap segala informasi yang baru. Kemudian kecerdasan buatan diharapkan dapat menyelesaikan masalah-masalah yang akan terjadi. Salah satu teknik kecerdasan buatan yang digunakan ialah *artificial neural network* (jaringan syaraf tiruan). Dinamakan *artificial neural network* karena meniru cara kerja dari jaringan saraf manusia yaitu menyampaikan input melalui neuron-neuron yang berkumpul membentuk suatu jaringan dan akhirnya keluar sebagai output.

Proses menerima input yang diteruskan kemudian oleh neuron dan menghasilkan suatu output memakai aplikasi pada graf. Sehingga sebelum memahami *artificial neural network* lebih lanjut dibutuhkan pemahaman akan topik graf. Terdapat banyak aplikasi *artificial neural network* diantaranya *image processing*, *pattern recognition*, *obat-obatan*, *speech recognition*, dan lain-lain.

Judul makalah ini dipilih karena merupakan topik yang menarik dalam abad ke-21 dimana teknologi, kecerdasan buatan, dan internet sudah berkembang dengan pesat. Selain itu *artificial neural network* dapat menyelesaikan sejumlah permasalahan dalam kehidupan manusia.

## II. LANDASAN TEORI

### 2.1 Teori Graf<sup>[1]</sup>

#### 2.1.1 Definisi Graf

Misalkan ada graf  $G = (V, E)$

$V$  merupakan himpunan simpul-simpul.

$E$  merupakan himpunan sisi-sisi.

$V$  tidak boleh kosong sedangkan  $E$  boleh kosong.

Sisi ganda dalam graf adalah dua sisi atau lebih yang menghubungkan dua simpul yang sama.

Gelang atau kalang adalah sisi yang mulai dan berakhir pada simpul yang sama.

#### 2.1.2 Jenis-Jenis Graf

Berdasarkan ada tidaknya gelang atau sisi ganda:

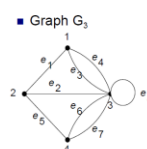
##### 1. Graf sederhana

Graf yang tidak mengandung gelang dan sisi-ganda (tidak memiliki keduanya).

##### 2. Graf tak-sederhana

Graf yang mengandung sisi ganda atau gelang (salah satu atau keduanya).

Graph



Pada  $G_3$ , sisi  $e_7 = (3, 3)$  dinamakan gelang atau kalang (*loop*) karena ia berawal dan berakhir pada simpul yang sama.

Gambar Graf tak-sederhana diambil dari [darkrabbitblog.blogspot.com](http://darkrabbitblog.blogspot.com)

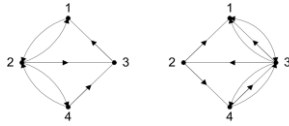
Berdasarkan ada tidaknya arah pada sisi:

##### 1. Graf tak-berarah

Graf yang sisi-sisinya tidak mempunyai arah.

## 2. Graf berarah

Graf yang sisi-sisinya mempunyai arah.



Gbr (a) graf berarah, (b) graf-ganda berarah

Gambar Graf berarah diambil dari soalampuhh.blogspot.com

### 2.1.3 Terminologi Graf

#### 1. Ketetangan (*Adjacent*)

Jika ada sisi yang menghubungkan dua buah simpul maka dua buah simpul tersebut merupakan tetangga.

#### 2. Bersisian (*Incidency*)

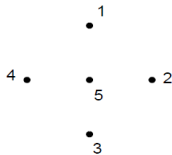
Jika ada sisi yang menghubungkan dua buah simpul maka sisi tersebut bersisian dengan dua buah simpul tersebut.

#### 3. Simpul Terpencil (*Isolated Vertex*)

Simpul yang tidak memiliki satu pun sisi yang bersisian dengannya.

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

Graf yang himpunan sisi-sisinya ( $E$ ) kosong.



Gambar graf kosong diambil dari rabbitjeyek.blogspot.com

#### 5. Derajat (*Degree*)

Jumlah sisi yang bersisian dengan simpul tersebut. Notasinya ialah  $d(v)$ . Terdapat suatu lemma jabat tangan dimana jumlah seluruh derajat semua simpul akan genap dimana jumlahnya ialah dua kali jumlah sisi graf tersebut. Sehingga akan mengakibatkan untuk sembarang graf maka jumlah simpul berderajat ganjil akan selalu genap (teorema).

#### 6. Lintasan (*Path*)

Suatu lintasan pada graf ialah barisan yang berisi nama dari simpul asal kemudian simpul berikutnya dan berikutnya hingga sampai di simpul tujuan. Panjang lintasan ialah jumlah sisi dalam lintasan tersebut.

#### 7. Siklus (*Cycle*) atau Sirkuit (*Circuit*)

Lintasan yang dimulai dan diakhiri dengan simpul yang sama.

## 8. Terhubung (*Connected*)

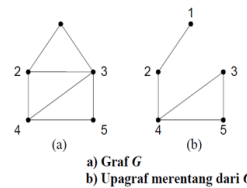
Jika dua buah simpul terdapat lintasan maka mereka terhubung. Jika graf terhubung maka untuk setiap simpul dalam himpunan  $V$  terdapat lintasannya. Jika tidak seluruhnya maka graf tersebut merupakan graf tak-terhubung.

## 9. Upagraf (*Subgraph*) dan Komplemen Upagraf

Jika ada graf  $G = (V, E)$  dan ada graf  $G_1 = (V_1, E_1)$   $G_1$  dikatakan upagraf jika  $V_1$  merupakan himpunan bagian dari  $V$  dan  $E_1$  merupakan himpunan bagian dari  $E$ . Komplemennya ialah suatu  $G_2$  dimana  $E_2 = E - E_1$  dan  $V_2$  adalah himpunan simpul yang anggota anggota  $E_2$  bersisian dengannya.

## 10. Upagraf Merentang (*Spanning Subgraph*)

Jika ada graf  $G = (V, E)$  dan ada graf  $G_1 = (V_1, E_1)$   $G_1$  dikatakan upagraf merentang jika  $V_1 = V$ .



a) Graf  $G$   
b) Upagraf merentang dari  $G$

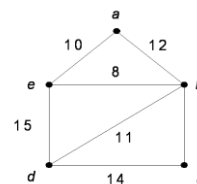
Gambar diambil dari adytia18.blogspot.com

## 11. *Cut-Set*

Terdapat suatu graf terhubung  $G$ , *cut-set* merupakan himpunan sisi yang jika dibuang dari  $G$  menyebabkan  $G$  menjadi tidak terhubung dan akan menyebabkan dua buah komponen.

## 12. Graf Berbobot (*Weighted Graph*)

Graf yang sisinya ada nilai atau bobotnya.

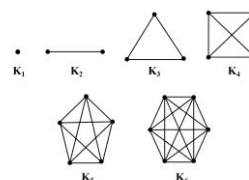


Gambar berbobot diambil dari sha-essa.blogspot.com

### 2.1.4 Graf Khusus

#### a. Graf Lengkap (*Complete Graph*)

Graf sederhana yang setiap simpulnya mempunyai sisi ke sisa simpul lain. Jumlah sisi pada graf ini yang terdiri dari  $n$  simpul adalah  $n(n-1)/2$ .



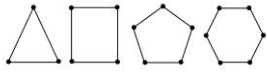
Gambar 12.20 Graf lengkap  $K_n$   
 $1 \leq n \leq 6$

homeomorfik (homeomorphic) dengan salah satu dari keduanya.

Graf Lengkap diambil dari  
slideplayer.info

#### b Graf Lingkaran

Graf sederhana yang tiap simpulnya mempunyai derajat sebesar dua.



Gambar 12.21 Graf lingkaran  $C_n$   
 $3 \leq n \leq 6$

Graf lingkaran diambil dari  
slideplayer.info

#### c. Graf Teratur (Regular Graph)

Graf yang setiap simpulnya mempunyai derajat yang sama disebut graf teratur. Apabila derajat setiap simpul adalah  $r$ , maka graf tersebut disebut sebagai graf teratur derajat  $r$ . Jumlah sisi pada graf teratur adalah  $nr/2$ .

#### d. Graf Bipartite (Bipartite Graph)

Graf  $G$  yang himpunan simpulnya dapat dipisah menjadi dua himpunan bagian  $V_1$  dan  $V_2$ , sedemikian sehingga setiap sisi pada  $G$  menghubungkan sebuah simpul di  $V_1$  ke sebuah simpul di  $V_2$  disebut graf bipartit dan dinyatakan sebagai  $G(V_1, V_2)$ .

### 2.1.5 Representasi Graf

1. Matriks Ketetangaan
2. Matriks Bersisian
3. Senarai Ketetangaan (*adjency list*)

### 2.1.6 Graf Isomorfik

Graf dikatakan isomorfik jika merupakan dua buah graf sama namun berbeda penggambarannya

### 2.1.7 Graf Planar dan Graf Bidang

Graf Planar adalah graf yang dapat digambarkan tanpa adanya sisi-sisi yang saling memotong sedangkan graf bidang merupakan graf planar yang telah dimodifikasi tanpa adanya sisi yang saling berpotongan.

### 2.1.8 Teorema Kuratowski

Bertujuan untuk menentukan apakah suatu graf planar atau tidak. Graf  $G$  bersifat planar jika dan hanya jika ia tidak mengandung upagraf yang isomorfik dengan salah satu graf Kuratowski atau

### 2.1.9 Lintasan dan Sirkuit Euler

Lintasan yang melalui masing-masing sisi dalam graf hanya satu kali secara tepat. Sedangkan sirkuit Euler adalah sirkuit yang melewati masing-masing sisi hanya satu kali saja secara tepat.

### 2.1.10 Lintasan dan Sirkuit Hamilton

Lintasan yang melalui masing-masing simpul dalam graf hanya satu kali secara tepat. Sedangkan sirkuit Euler adalah sirkuit yang melewati masing-masing simpul hanya satu kali saja secara tepat kecuali simpul asal.

## 2.2 Teori *Artificial Neural Network*<sup>[2][3][5]</sup>

### 2.2.1 Sejarah *Artificial Neural Network*

*Artificial intelligence* atau kecerdasan buatan sedang maraknya berkembang untuk meniru kepintaran manusia. Kemudian pertama-tama diadakan suatu studi untuk mengenal teori dasar mekanisme proses terjadinya intelegensi yang disebut *Cognitive Science*. Lalu dari teori dasar tersebut dibentuk suatu model yang akan disimulasikan pada komputer yang dalam perkembangannya menjadi *artificial intelligence*. *Artificial neural network* merupakan salah satu bidang dibagian *artificial intelligence* tersebut. *Artificial neural network* merupakan suatu ilmu yang relatif baru. *Artificial neural network* bermula pada makalah Waffen McCulloch dan Walter Pitts pada tahun 1943. Pada makalah tersebut mereka mencoba untuk menformulasi sel-sel otak menjadi model matematis. Metode yang dikembangkan mereka dikembangkan lebih lanjut dan merupakan suatu langkah maju di dalam perkembangan informatika.

### 2.2.2 Pengertian *Artificial Neural Network*

*Artificial neural network* adalah sistem komputasi dengan operasi dan arsitektur yang mirip dengan sel saraf biologis manusia di dalam otak. *Artificial neural network* merupakan salah satu representasi buatan otak manusia. *Artificial neural network* dapat digunakan untuk memodelkan hubungan yang kompleks antara input dan output untuk menemukan pola-pola data.

### 2.2.3 Konsep Dasar *Artificial Neural Network*

1. Pemrosesan informasi terjadi pada beberapa *neuron* (sebuah unit pemroses informasi yang merupakan dasar operasi *artificial neural network*).
2. Sinyal yang dilewatkan antara *neuron* akan menciptakan jaringan koneksi
3. Setiap koneksi penghubung memiliki bobot yang terhubung yang dalam jaringan saraf tertentu akan mengalikan sinyal yang ditransmisikan.

4. Setiap *neuron* mempunyai fungsi aktivasi yang non linier pada jaringan inputnya untuk menentukan sinyal output.
5. Adaptasi dan pelatihan diperlukan pada suatu *artificial neural network* sebelum siap digunakan untuk menyelesaikan suatu masalah. Lalu *artificial neural network* dapat memberikan solusi yang benar terhadap masalah-masalah yang ada.

### 2.2.4 Karakteristik Artificial Neural Network

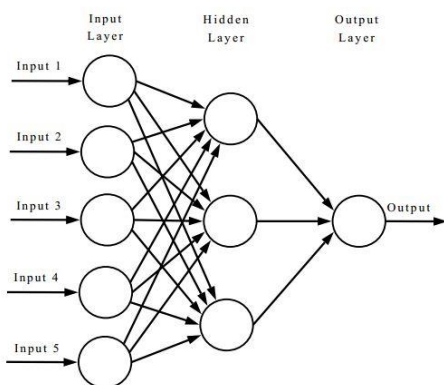
Penyelesaian masalah dengan jaringan syaraf tiruan tidak memerlukan pemrograman. Jaringan syaraf tiruan menyelesaikan masalah melalui proses belajar dari contoh-contoh pelatihan yang diberikan. Biasanya pada jaringan syaraf tiruan diberikan sebuah himpunan pola pelatihan yang terdiri dari sekumpulan contoh pola. Proses belajar jaringan syaraf tiruan berasal dari serangkaian contoh-contoh pola yang diberikan. metode pelatihan yang sering dipakai adalah metode belajar terbimbing. Selama proses belajar itu pola masukan disajikan bersama-sama dengan pola keluaran yang diinginkan. Jaringan akan menyesuaikan nilai bobotnya sebagai tanggapan atas pola masukan dan sasaran yang disajikan tersebut.

### 2.2.5 Definisi Algoritma Backpropagation

*Backpropagation* adalah suatu algoritma pembelajaran yang terwariskan dan digunakan oleh *perceptron* dengan banyak lapisan untuk mengubah bobot-bobot yang terhubung dengan neuron-neuron yang ada pada lapisan tersembunyinya. Algoritma *backpropagation* menggunakan *error output* untuk mengubah nilai-nilai bobotnya dalam arah mundur (*backward*). Tahap perambatan maju (*forward propagation*) harus dikerjakan terlebih dahulu untuk mendapatkan nilai error tersebut.

## III. APLIKASI GRAF PADA ARTIFICIAL NEURAL NETWORK

### 3.1. Penerimaan Input, Pemrosesan, dan Output yang dihasilkan dari jaringan kecil (neuron).



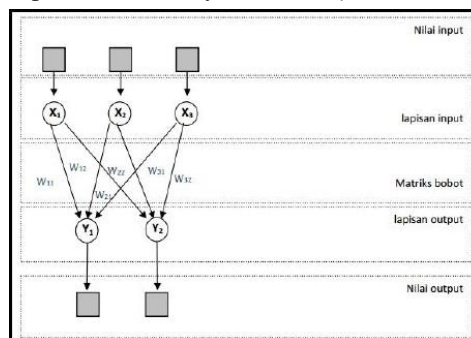
Gambar diambil dari [tex.stackexchange.com](http://tex.stackexchange.com)

*Artificial neural network* tersusun atas jaringan kecil (*neuron*) yang terbagi atas *neuron* input, *neuron* yang tersembunyi (pemroses), *neuron* output sebagai simpul, serta koneksi atau jaringan yang menghubungkan dari input, tersembunyi, dan output sebagai sisi dari graf. Tipe graf dari bentuk tersebut ialah graf yang mempunyai orientasi arah dan merupakan graf berbobot karena setiap sisi akan ada bobotnya tersendiri. Berikut akan dijelaskan mengenai jenis arsitektur yang memanfaatkan aplikasi graf.

### 3.2. Jenis Arsitektur Artificial Neural Network

#### 3.2.1 Lapisan tunggal (single layer net)

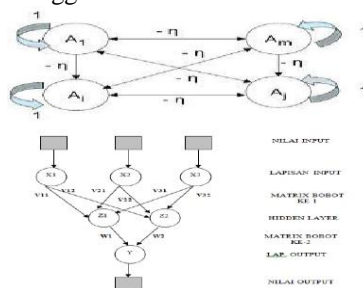
Jaringan ini mempunyai hanya satu lapisan dengan bobot yang terhubung. Jaringan ini menerima input dan kemudian secara langsung akan mengolah menjadi output tanpa adanya lapisan tersembunyi (*hidden layer*).



Gambar diambil dari [ulfarieantifst10.web.unair.ac.id](http://ulfarieantifst10.web.unair.ac.id)

#### 3.2.2 Jaringan dengan banyak lapisan (multilayer)

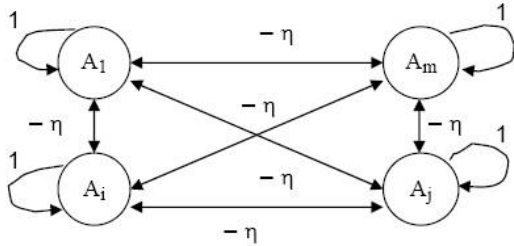
Jaringan ini mempunyai satu atau lebih lapisan yang terletak diantara lapisan input dan outputnya. Jaringan bisa mempunyai satu atau lebih lapisan yang tersembunyi. Jaringan ini berguna untuk menyelesaikan masalah yang sulit daripada jaringan dengan lapisan tunggal. Akan tetapi pembelajaran jaringan akan lebih rumit jika dibandingkan dengan jaringan lapisan tunggal. Keberhasilan jaringan ini pun lebih tinggi jika dibandingkan dengan jaringan lapisan tunggal.



Gambar diambil dari [ulfarieantifst10.web.unair.ac.id](http://ulfarieantifst10.web.unair.ac.id)

### 3.2.3 Jaringan dengan lapisan kompetitif (*competitive layer net*)

Pada jaringan ini seluruh neuron akan bersaing untuk mendapatkan suatu hak aktif.



Gambar diambil dari [webinside.esy.es](http://webinside.esy.es)

### 3.3. Fungsi Aktivasi

Fungsi aktivasi dipakai untuk menentukan output dari suatu neuron. Merupakan suatu fungsi yang menggambarkan hubungan antara tingkat aktivasi internal (*summation function*) yang mungkin berbentuk linier atau non-linier.

### 3.4. Model Artificial Neural Network Backpropagation

Pelatihan pada jaringan syaraf *backpropagation*, *feedforward* dilakukan dengan perhitungan bobot sehingga pada akhir pelatihan akan diperoleh bobot-bobot yang baik. Selama proses pelatihan, bobot-bobot diatur secara iteratif untuk meminimumkan *error* yang terjadi. *Error* dihitung berdasarkan rata-rata kuadrat kesalahan. Rata-rata kuadrat kesalahan juga dijadikan dasar perhitungan unjuk kerja fungsi aktivasi. Sebagian besar pelatihan untuk jaringan *feedforward* menggunakan gradien dari fungsi aktivasi untuk menentukan bagaimana mengatur bobot-bobot dalam rangka meminimumkan kinerja. Gradien ini ditentukan dengan menggunakan suatu teknik yang disebut *backpropagation*.

Algoritma pelatihan standar *backpropagation* akan menggerakkan bobot dengan arah gradien negatif. Prinsip dasar dari algoritma *backpropagation* adalah memperbaiki bobot-bobot jaringan dengan arah yang membuat fungsi aktivasi menjadi turun dengan cepat.

Pelatihan *backpropagation* meliputi 3 tahapan sebagai berikut.

#### 1. Propagasi maju.

Pola masukan dihitung maju mulai dari *input layer* hingga *output layer* menggunakan fungsi aktivasi yang ditentukan.

#### 2. Propagasi mundur.

Selisih antara keluaran jaringan dengan target yang diinginkan merupakan kesalahan yang terjadi. Kesalahan yang terjadi itu dipropagasi mundur.

Dimulai dari garis yang berhubungan langsung dengan unit-unit di *output layer*

#### 3. Perubahan bobot.

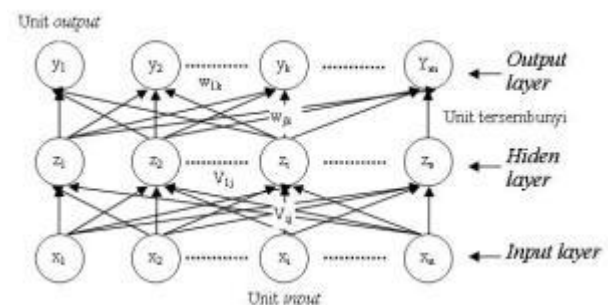
Modifikasi bobot untuk menurunkan kesalahan yang terjadi. Ketiga fase tersebut diulang-ulang terus hingga kondisi penghentian dipenuhi.

### 3.5. Algoritma Artificial Neural Network Backpropagation<sup>[4]</sup>

Saat perambatan koneksi atau jaringan neuron-neuron diaktifkan terdapat suatu fungsi aktivasi sigmoid biner sebagai berikut :

$$f(x) = \frac{1}{1 + e^{-x}}$$

Dengan arsitektur *artificial neural network backpropagation* seperti gambar berikut :



Gambar diambil dari [javaneural.blogspot.com](http://javaneural.blogspot.com)

#### Algoritma *backpropagation*:

1. Inisialisasi bobot (ambil bobot awal dengan nilai random yang cukup kecil).

2. Kerjakan langkah-langkah berikut selama kondisi berhenti belum terpenuhi.

Tiap-tiap pasangan elemen yang akan dilakukan pembelajaran, lakukan:

#### *Feedforward*

1. Tiap unit input menerima sinyal dan meneruskan sinyal tersebut pada lapisan yang ada di atasnya (lapisan tersembunyi).

2. Tiap unit tersembunyi menjumlahkan sinyal-sinyal input terbobot:

$$z\_in_j = v_0j + \sum_{i=1}^n x_i v_{ij}$$

Gambar diambil dari javaneural.blogspot.com

gunakan fungsi aktivasi untuk menghitung sinyal outputnya:  $z_j = f(z_{in_j})$  dan kirimkan sinyal tersebut ke semua unit dilapisan atasnya (unit-unit output).

3. Tiap unit output menjumlahkan sinyal-sinyal input terbobot

$$y_{in_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk}$$

Gambar diambil dari javaneural.blogspot.com

gunakan fungsi aktivasi untuk menghitung sinyal outputnya  $y_k = f(y_{in_k})$  dan kirimkan sinyal tersebut ke semua unit dilapisan atasnya (unit-unit output).

*Backpropagation*

4. Tiap-tiap unit output menerima target pola yang berhubungan dengan pola input pembelajaran. Hitung informasi errornya dengan rumus  $\sigma_k = (t_k - y_k) f'(y_{in_k})$ . Kemudian hitung koreksi bobot dengan rumus  $\Delta w_{jk} = \alpha \sigma_k z_{ij}$ . Hitung juga koreksi bias:  $\Delta v_{jk} = \alpha \sigma_k$ . Kirimkan  $\sigma_k$  ini ke unit-unit yang ada dilapisan bawahnya.

5. Tiap-tiap unit tersembunyi menjumlahkan delta inputnya (dari unit-unit yang berada pada lapisan di atasnya):

$$\sigma_{in_j} = \sum_{k=1}^m \sigma_k w_{jk}$$

Gambar diambil dari javaneural.blogspot.com

kalikan nilai ini dengan turunan dari fungsi aktivasinya untuk menghitung informasi *error*:  $\sigma_j = \sigma_{in_j} f'(z_{in_j})$ . Kemudian hitung koreksi bobot (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{ij}$ ):  $\Delta v_{jk} = \alpha \sigma_j x_i$ . Hitung juga koreksi bias (yang nantinya akan digunakan untuk memperbaiki nilai  $v_{0j}$ ):  $\Delta v_{0j} = \alpha \sigma_j$ . Tiap unit output memperbaiki bias dan bobotnya.  $w_{jk}(\text{baru}) = w_{jk}(\text{lama}) + \Delta w_{jk}$

Tiap-tiap unit tersembunyi memperbaiki bias dan bobotnya.

$$v_{ij}(\text{baru}) = v_{ij}(\text{lama}) + \Delta v_{ij}$$

Kemudian terakhir tes kondisi berhenti.

## IV. KESIMPULAN

Aplikasi graf banyak diterapkan dalam bidang informatika. Salah satunya ialah *artificial neural network* yang merupakan cabang dari *artificial intelligent*. Penerapan graf pada *artificial neural network* adalah *neuron* input, *neuron* output, *hidden neuron* atau *neuron pemroses* sebagai simpulnya dan jaringan atau koneksi dari *neuron* terhubung sebagai sisi dengan sisi yang memiliki bobot. Fungsi dari *artificial neural network* ialah membantu kehidupan manusia untuk menyelesaikan masalah-masalah yang manusia hadapi secara tepat, cepat, dan tanggap serta sistem yang dapat belajar secara terus menerus seperti halnya seorang manusia. Algoritma *backpropagation* merupakan algoritma yang berfungsi untuk mengubah bobot-bobot atau sisi dari graf pada *neuron* untuk mencari bobot yang baik dan tepat untuk seluruh *artificial neural network* dengan memanfaatkan aplikasi graf berbobot. Dengan semakin berkembangnya kecerdasan buatan diharapkan dapat membantu manusia dalam menyelesaikan permasalahan-permasalahan yang ada.

## V. UCAPAN TERIMA KASIH

Pertama-tama penulis mengucapkan terima kasih kepada Tuhan Yang Maha Esa oleh karena anugerah-Nya penulis dapat menyelesaikan semua tulisan ini. Penulis ingin berterima kasih kepada dosen IF 2120 yaitu Pak Rinaldi Munir dan Ibu Harlili. Serta penulis juga mengucapkan terima kasih yang tidak terhingga kepada semua teman-teman seperjuangan yang membantu penulis untuk menyelesaikan tulisan ini. Penulis pun tidak lupa mengucapkan terima kasih kepada semua pembaca tulisan ini dan semoga tulisan ini dapat bermanfaat bagi para pembacanya.

## REFERENSI

- [1] informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2015-2016/Graf%20(2015).ppt 5 Desember 2015 pada jam 21:14
- [2] <http://fridusuhendra.blogspot.co.id/2013/09/jaringan-syaraf-tiruan-jaringan-syaraf.html> 5 Desember 2015 pada jam 23:02
- [3] [http://www.academia.edu/9912265/Jaringan\\_Syaraf\\_Tiruan](http://www.academia.edu/9912265/Jaringan_Syaraf_Tiruan) 5 Desember 2015 pada jam 23:44
- [4] <http://javaneural.blogspot.co.id/2009/11/algoritma-pembelajaran-backpropagation.html> 6 Desember 2015 pada jam 19:56
- [5] <https://jalanwaktu.wordpress.com/jaringan-syaraf-tiruan/> 6 Desember 2015 pada jam 23:50
- [6] Hermawan, A. (2006). *Jaringan Syaraf Tiruan Teori dan Aplikasi*. Yogyakarta: Penerbit Andi
- [7] Gurney, K. (1997) *An Introduction to Neural Networks* London: Routledge
- [8] Puspasingrum, D. (2006). *Pengantar Jaringan Syaraf Tiruan*. Yogyakarta : Andi Offset

## 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, 8 Desember 2015

A handwritten signature in black ink, appearing to read 'Joshua Salimin', with a horizontal line underneath the name.

Joshua Salimin  
13514001