

# **BAB IV**

## **JARINGAN SYARAF TIRUAN**

### **(*ARTIFICIAL NEURAL NETWORK*)**

#### **Kompetensi :**

1. Mahasiswa memahami konsep Jaringan Syaraf Tiruan

#### **Sub Kompetensi :**

1. Dapat mengetahui sejarah JST
2. Dapat mengetahui macam-macam arsitektur JST
3. Dapat mengetahui macam-macam fungsi aktivasi JST
4. Dapat mengetahui macam-macam algoritma dalam JST

#### **IV.1. Pendahuluan**

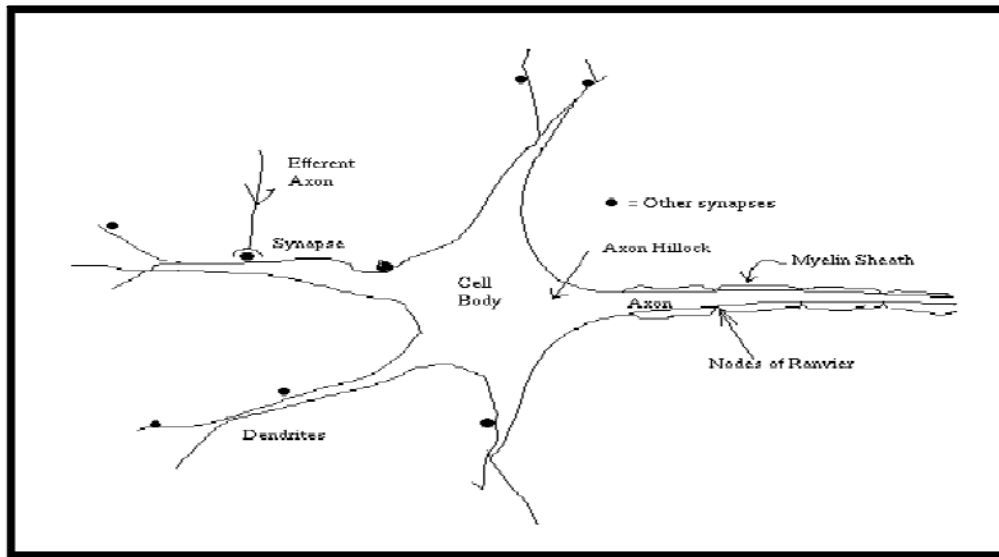
Jaringan Syaraf Tiruan dibuat pertama kali pada tahun 1943 oleh *neurophysiologist* Warren McCulloch dan *logician* Walter Pitts, namun teknologi yang tersedia pada saat itu belum memungkinkan mereka berbuat lebih jauh.

Jaringan Syaraf Tiruan adalah paradigma pemrosesan suatu informasi yang terinspirasi oleh sistem sel syaraf biologi, sama seperti otak yang memproses suatu informasi. Elemen mendasar dari paradigma tersebut adalah struktur yang baru dari sistem pemrosesan informasi. Jaringan Syaraf Tiruan, seperti manusia, belajar dari suatu contoh. Jaringan Syaraf Tiruan dibentuk untuk memecahkan suatu masalah tertentu seperti pengenalan pola atau klasifikasi karena proses pembelajaran. Jaringan Syaraf Tiruan berkembang secara pesat pada beberapa tahun terakhir.

Jaringan Syaraf Tiruan telah dikembangkan sebelum adanya suatu computer konvensional yang canggih dan terus berkembang walaupun pernah mengalami masa vakum selama beberapa tahun.

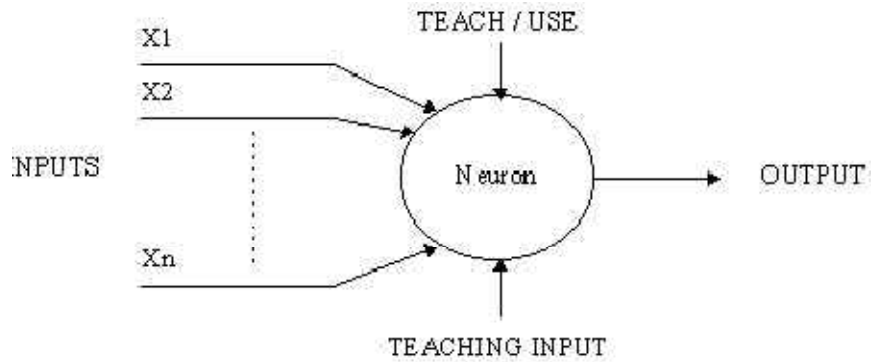
Jaringan Syaraf Tiruan keluar dari penelitian kecerdasan buatan, terutama percobaan untuk menirukan *fault-tolerance* dan kemampuan untuk belajar dari sistem syaraf biologi dengan model struktur *low-level* dari otak. Otak terdiri dari sekitar (10.000.000.000) sel syaraf yang saling berhubungan. Sel syaraf mempunyai cabang struktur input (dendrites), sebuah inti sel dan percabangan struktur output (axon). Axon dari sebuah sel terhubung dengan dendrites yang lain melalui sebuah synapse. Ketika sebuah sel syaraf aktif, kemudian menimbulkan suatu signal

electrochemical pada axon. Signal ini melewati synapses menuju ke sel syaraf yang lain. Sebuah sel syaraf lain akan mendapatkan signal jika memenuhi batasan tertentu yang sering disebut dengan nilai ambang atau (threshold).



*Gambar 1. Susunan Syaraf Manusia*

Jaringan Syaraf Tiruan memiliki pendekatan yang berbeda untuk memecahkan masalah bila dibandingkan dengan sebuah komputer konvensional. Umumnya komputer konvensional menggunakan pendekatan algoritma (komputer konvensional menjalankan sekumpulan perintah untuk memecahkan masalah). Jika suatu perintah tidak diketahui oleh komputer konvensional maka komputer konvensional tidak dapat memecahkan masalah yang ada. Sangat penting mengetahui bagaimana memecahkan suatu masalah pada komputer konvensional dimana komputer konvensional akan sangat bermanfaat jika dapat melakukan sesuatu dimana pengguna belum mengetahui bagaimana melakukannya. Jaringan Syaraf Tiruan dan suatu algoritma komputer konvensional tidak saling bersaing namun saling melengkapi satu sama lain. Pada suatu kegiatan yang besar, sistem yang diperlukan biasanya menggunakan kombinasi antara keduanya (biasanya sebuah komputer konvensional digunakan untuk mengontrol Jaringan Syaraf Tiruan untuk menghasilkan efisiensi yang maksimal. Jaringan Syaraf Tiruan tidak memberikan suatu keajaiban tetapi jika digunakan secara tepat akan menghasilkan sesuatu hasil yang luarbiasa.



Gambar 2. Sebuah Sel Syaraf Sederhana

JST dibentuk sebagai generalisasi model matematika dari jaringan syaraf biologi, dengan asumsi bahwa:

- ✓ Pemrosesan informasi terjadi pada banyak elemen sederhana (neuron).
- ✓ Sinyal dikirimkan diantara neuron-neuron melalui penghubung-penghubung.
- ✓ Penghubung antar neuron memiliki bobot yang akan memperkuat atau memperlemah sinyal.

#### IV.2. Sejarah JST

Jaringan syaraf tiruan sederhana pertama kali diperkenalkan oleh McCulloch dan Pitts di tahun 1943. McCulloch dan Pitts menyimpulkan bahwa kombinasi beberapa *neuron* sederhana menjadi sebuah sistem neural akan meningkatkan kemampuan komputasinya. Bobot dalam jaringan yang diusulkan oleh McCulloch dan Pitts diatur untuk melakukan fungsi logika sederhana. Fungsi aktivasi yang dipakai adalah fungsi *threshold*. Tahun 1958, Rosenblatt memperkenalkan dan mulai mengembangkan model jaringan yang disebut Perceptron. Metode pelatihan diperkenalkan untuk mengoptimalkan hasil iterasinya.

Widrow dan Hoff (1960) mengembangkan perceptron dengan memperkenalkan aturan pelatihan jaringan, yang dikenal sebagai *aturan delta* (atau sering disebut kuadrat rata-rata terkecil). Aturan ini akan mengubah bobot perceptron apabila keluaran yang dihasilkan tidak sesuai dengan target yang diinginkan.

Selain itu, beberapa model jaringan syaraf tiruan lain juga dikembangkan oleh Kohonen (1972), Hopfield (1982), dll. Pengembangan yang ramai dibicarakan sejak tahun 1990an adalah aplikasi model-model jaringan syaraf tiruan untuk menyelesaikan berbagai masalah di dunia nyata.

Jaringan Syaraf Tiruan ditentukan oleh tiga hal :

- Pola hubungan antar neuron (disebut arsitektur jaringan)
- Metode untuk menentukan bobot penghubung (disebut metode learning/training)
- Fungsi aktivasi

### **IV.3. Arsitektur Jaringan**

Beberapa arsitektur jaringan yang sering dipakai dalam jaringan syaraf tiruan antara lain :

#### 1. Jaringan Lapisan Tunggal (single layer network)

Dalam jaringan ini, sekumpulan input neuron dihubungkan langsung dengan sekumpulan outputnya. Dalam beberapa model (misal perceptron), hanya ada sebuah unit neuron output.

#### 2. Jaringan Lapisan Jamak (multi layer network)

Jaringan layar jamak merupakan perluasan dari layar tunggal. Dalam jaringan ini, selain unit input dan output, ada unit-unit lain (sering disebut layar tersembunyi / hidden layer). Dimungkinkan pula ada beberapa layar tersembunyi. Sama seperti pada unit input dan output, unit-unit dalam satu layar tidak saling berhubungan.

#### 3. Jaringan Recurrent

Model jaringan reccurent mirip dengan jaringan layar tunggal ataupun ganda. Hanya saja, ada neuron output yang memberikan sinyal pada unit input (sering disebut feedback loop).

### **IV.4. Paradigma Pembelajaran dalam JST**

Berdasarkan cara memodifikasi bobotnya, ada 2 macam pelatihan yang dikenal yaitu dengan supervisi dan tanpa supervisi. Dalam pelatihan dengan supervisi, terdapat sejumlah pasangan data (masukan - target keluaran) yang dipakai untuk melatih jaringan hingga diperoleh bobot yang diinginkan. Pasangan data tersebut berfungsi sebagai "guru" untuk melatih jaringan hingga diperoleh bentuk yang terbaik. "Guru" akan memberikan informasi yang jelas tentang bagaimana sistem harus mengubah dirinya untuk meningkatkan unjuk kerjanya. Pada setiap kali pelatihan, suatu input diberikan ke jaringan. Jaringan akan memproses dan mengeluarkan keluaran. Selisih antara keluaran jaringan dengan target (keluaran yang diinginkan) merupakan kesalahan yang terjadi.

Jaringan akan memodifikasi bobot sesuai dengan kesalahan tersebut. Jaringan perceptron, Adaline dan backpropagation merupakan model-model yang menggunakan pelatihan dengan supervisi. Sebaliknya, dalam pelatihan tanpa supervisi tidak ada "guru" yang akan mengarahkan proses pelatihan. Dalam pelatihannya, perubahan bobot jaringan dilakukan berdasarkan parameter tertentu dan jaringan dimodifikasi menurut ukuran parameter tersebut.

Sebagai contoh, dalam model jaringan kompetitif, jaringan terdiri dari 2 layar, yaitu layar input dan layar kompetisi. Layar input menerima data eksternal. Layar kompetitif berisi neuron-neuron yang saling berkompetisi agar memperoleh kesempatan untuk merespon sifat-sifat yang ada dalam data masukan. Neuron yang memenangkan kompetisi akan memperoleh sinyal yang berikutnya ia teruskan. Bobot neuron pemenang akan dimodifikasi sehingga lebih menyerupai data masukan.

Sebagai ilustrasi, pelatihan dengan supervisi dapat diandaikan sebagai skripsi yang dibimbing oleh seorang dosen. Pada setiap kali pengumpulan berkas skripsi, dosen akan mengkritik, mengarahkan dan meminta perbaikan agar kualitas skripsi meningkat. Sebaliknya, Dalam pelatihan tanpa supervisi dapat dibayangkan sebagai skripsi tanpa dosen pembimbing. Mahasiswa mengerjakan skripsi sebaik-baiknya berdasarkan ukuran tertentu (misal dibandingkan dengan skripsi yang sudah ada sebelumnya atau dibandingkan dengan hasil skripsi temannya). Berdasarkan hasil yang pernah dilaporkan, model pelatihan dengan supervisi lebih banyak digunakan dan terbukti cocok dipakai dalam berbagai aplikasi. Akan tetapi kelemahan utama pelatihan dengan supervisi adalah dalam hal pertumbuhan waktu komputasinya yang berorde eksponensial. Ini berarti untuk data pelatihan yang cukup banyak, prosesnya menjadi sangat lambat.

## **1. Pelatihan dengan Supervisi**

Jaringan memberikan tanggapan dengan mendapatkan target tertentu. Sebelum jaringan mengubah sendiri bobotnya untuk mencapai target, bobot interkoneksi diinisialisasi. Proses belajar JST dengan pengawasan adalah proses belajar dengan memberikan latihan untuk mencapai suatu target keluaran yang ditentukan. JST mendapatkan latihan untuk mengenal pola-pola tertentu. Dengan memberikan target keluaran, perubahan masukan akan diadaptasi oleh keluaran dengan mengubah bobot interkoneksinya mengikuti algoritma belajar yang ditentukan. Set pelatihan dipilih dari fungsi keluaran maksimum setiap keadaan parameter yang diubah. Dengan menginisialisasi bobot tiap sel, JST

akan mencari error terkecil, sehingga bentuk fungsi keluaran mendekati target yang diinginkan. Berdasarkan proses belajar yang dilakukan.

Kita perlu memperhatikan beberapa hal dalam menyusun set pelatihan, yaitu :

- a. Pemberian urutan pola yang akan diajarkan
- b. Kriteria perhitungan error
- c. Kriteria proses belajar
- d. Jumlah iterasi yang harus dilalui
- e. Inisialisasi bobot dan parameter awal

Pelatihan dilakukan dengan memberikan pasangan pola-pola masukan dan keluaran. Untuk keperluan pengendalian, pasangan pola tidak mengikuti rumusan tertentu. JST harus dapat mengadaptasi masukan yang acak supaya keluaran tetap mengikuti target. Lebih lanjut, proses pelatihan dilakukan dengan memberikan pola yang menggunakan masukan acak dan bobot interkoneksi yang besar. Dengan pemberian bobot yang besar, perbedaan target dan keluaran berkurang lebih cepat, sehingga proses adaptasi akan lebih cepat pula. Salah satu proses belajar dengan pengawasan adalah proses belajar menggunakan algoritma propagasi balik. Proses belajar jaringan umpan balik dituliskan dalam bentuk algoritma propagasi balik yang dikenal sebagai JPB. Jaringan Propagasi Balik (JPB) kadang-kadang dikenal sebagai Multilayer Perceptron (MLP). Anda dapat menggunakan algoritma propagasi balik untuk melatih jaringan lapis banyak.

## **2. Pelatihan Tanpa Supervisi**

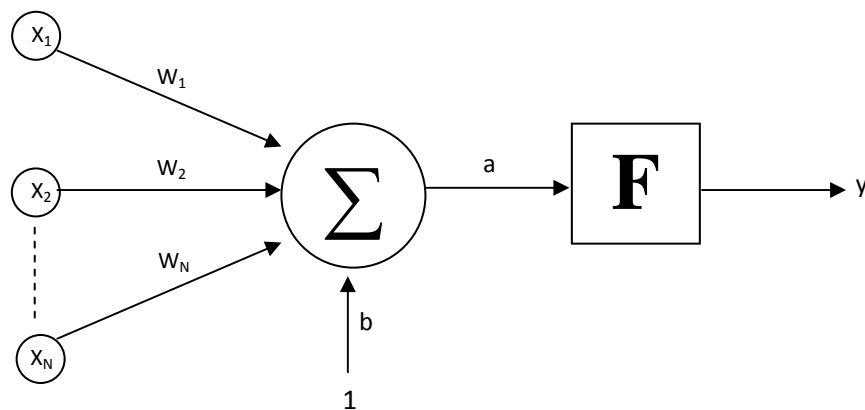
Pada pelatihan tanpa supervisi jaringan tidak mendapatkan target, sehingga JST mengatur bobot interkoneksi sendiri. Belajar tanpa pengawasan kadang-kadang diacu sebagai *self-organizing learning*, yakni belajar mengklasifikasikan tanpa dilatih. Pada proses belajar tanpa pengawasan, JST akan mengklasifikasikan contoh pola-pola masukan yang tersedia ke dalam kelompok yang berbeda-beda.

## IV.5. Fungsi Aktivasi

Fungsi aktivasi merupakan fungsi yang digunakan pada jaringan syaraf untuk mengaktifkan atau tidak mengaktifkan neuron. Seperti terlihat pada gambar 3 sebuah neuron akan mengolah N input ( $x_1, x_2, \dots, x_N$ ) yang masing-masing memiliki bobot  $w_1, w_2, w_3, \dots, w_N$  dan bobot bias  $b$ , dengan rumus :

$$a = \sum_{i=1}^N x_i \cdot w_i$$

Kemudian fungsi aktivasi  $F$  akan mengaktifkan  $a$  menjadi output jaringan  $y$ .



Gambar 3. Fungsi Aktivasi Pada Jaringan Syaraf Sederhana

Ada beberapa fungsi aktivasi yang sering digunakan dalam jaringan syaraf tiruan. Fungsi aktivasi yang disediakan pada toolbox matlab, antara lain:

### 1. Fungsi undak biner (*hardlim*)

Fungsi undak biner (step function) sering digunakan pada jaringan dengan lapisan tunggal untuk mengkonversi input dari suatu variabel yang bernilai kontinu ke suatu output biner (0 atau 1) dengan syntax:  $Y = \text{hardlim}(a)$ .

$$Y = \begin{cases} 0, & \text{jika } x \leq 0 \\ 1, & \text{jika } x > 0 \end{cases}$$

### 2. Fungsi bipolar (*hardlims*)

Fungsi bipolar sebenarnya hampir sama dengan fungsi undak biner, hanya saja output yang dihasilkan berupa 1 atau  $-1$ . Pada matlab fungsi ini lebih dikenal dengan nama **hardlims**, dengan syntax:  $Y = \text{hardlim}(a)$ .

$$Y = \begin{cases} 1, & \text{jika } x > 0 \\ 0, & \text{jika } x = 0 \\ -1, & \text{jika } x < 0 \end{cases}$$

### 3. Fungsi linier (purelin)

Fungsi linier memiliki nilai output yang sama dengan nilai inputnya. Pada matlab fungsi aktivasi linier (identitas) dikenal dengan nama **purelin**, dengan syntax :  $Y = \text{purelin}(a)$ .

$$\text{Dirumuskan : } y = x$$

### 4. Fungsi saturating linier (satlin)

Fungsi ini akan bernilai 0 jika inputnya kurang dari  $-\frac{1}{2}$ , dan akan bernilai 1 jika inputnya lebih dari  $\frac{1}{2}$ . Sedangkan jika nilai input terletak antara  $-\frac{1}{2}$  dan  $\frac{1}{2}$ , maka outputnya akan bernilai sama dengan nilai input ditambah  $\frac{1}{2}$ . Pada matlab syntax yang digunakan adalah:  $Y = \text{satlin}(a)$ .

$$Y = \begin{cases} 1; & \text{jika } x \geq 0 \\ x + 0,5; & \text{jika } -0,5 \leq x \leq 0,5 \\ 0; & \text{jika } x \leq -0,5 \end{cases}$$

### 5. Fungsi symmetric saturating linier (satlins)

Fungsi ini akan bernilai  $-1$  jika inputnya kurang dari  $-1$ , dan akan bernilai 1 jika inputnya lebih dari 1. Sedangkan jika nilai input terletak antara  $-1$  dan 1, maka outputnya akan bernilai sama dengan nilai inputnya. Pada matlab syntax yang digunakan adalah:  $Y = \text{satlins}(a)$ .

$$Y = \begin{cases} 1; & \text{jika } x \geq 1 \\ x; & \text{jika } -1 \leq x \leq 1 \\ -1; & \text{jika } x \leq -1 \end{cases}$$

### 6. Fungsi sigmoid biner (logsig)

Fungsi ini digunakan untuk jaringan syaraf yang dilatih dengan menggunakan *metode backpropagation*. Fungsi sigmoid biner memiliki nilai pada range 0 sampai 1. Oleh karena itu, fungsi ini sering digunakan untuk jaringan syaraf yang membutuhkan nilai output yang terletak pada interval 0 sampai 1, namun fungsi ini juga dapat digunakan oleh jaringan syaraf yang nilai outputnya 0 atau 1. Syntax yang digunakan pada matlab adalah:  $Y = \text{logsig}(a)$ .

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



### 7. Fungsi sigmoid bipolar (tansig)

Fungsi sigmoid bipolar hampir sama dengan fungsi sigmoid biner, hanya saja output dari fungsi ini memiliki range antara 1 sampai -1. Syntax yang digunakan pada matlab adalah:  $Y = \text{tansig}(a)$ .

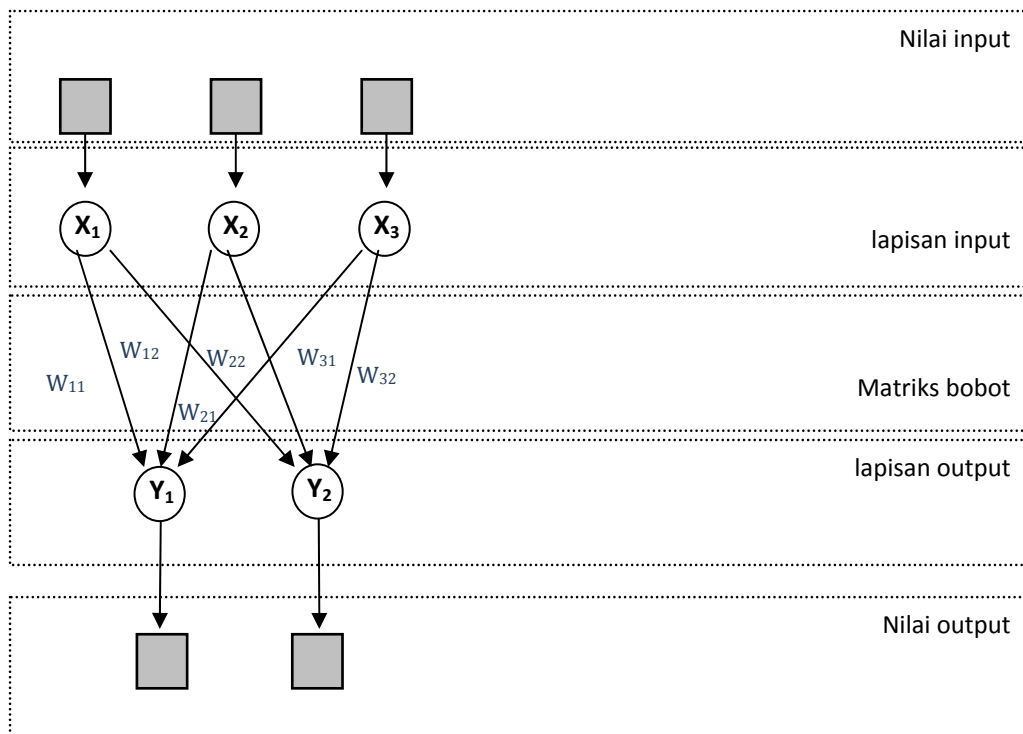
$$y = f(x) = \frac{1 - e^{-x}}{1 + e^{-x}}$$

## IV.6. Arsitektur JST

Beberapa arsitektur jaringan yang sering dipakai dalam jaringan syaraf tiruan antara lain:

### 1. Jaringan dengan lapisan tunggal (*single layer net*)

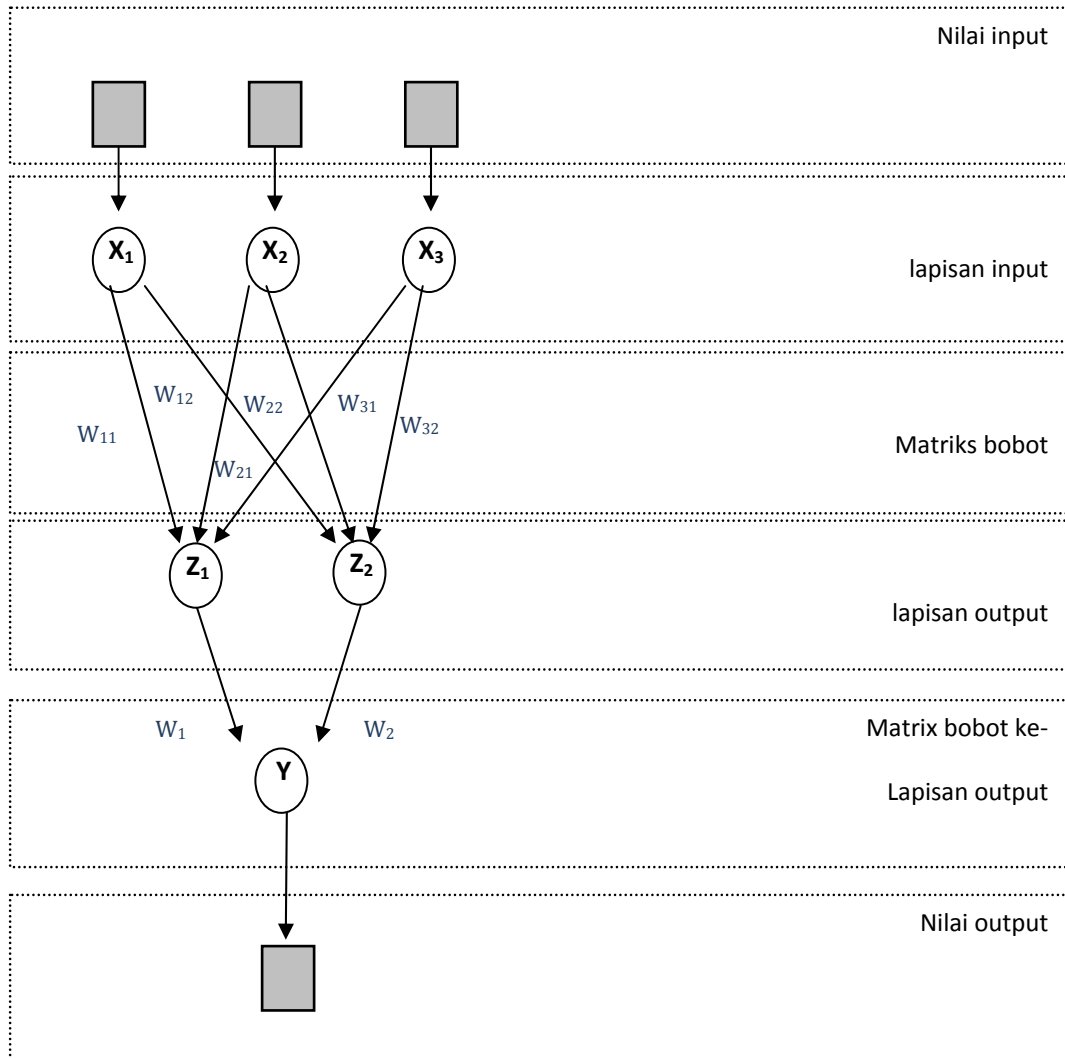
Jaringan dengan lapisan tunggal hanya memiliki satu lapisan dengan bobot-bobot terhubung. Jaringan ini hanya menerima input kemudian secara langsung akan mengolahnya menjadi output tanpa harus melalui lapisan tersembunyi. Pada gambar 4 menunjukkan jaringan syaraf dengan lapisan input memiliki 3 neuron, yaitu  $X_1$ ,  $X_2$ , dan  $X_3$ . Sedangkan pada lapisan output memiliki 2 neuron yaitu  $Y_1$  dan  $Y_2$ . Neuron-neuron pada kedua lapisan saling berhubungan, besar hubungan antara 2 neuron ditentukan oleh bobot yang bersesuaian.



Gambar 4. Jaringan Syaraf Tiruan Dengan Lapisan Tunggal Tunggal

## 2. Jaringan dengan banyak lapisan (*multi layer net*)

Jaringan dengan banyak lapisan memiliki satu atau lebih lapisan yang terletak diantara lapisan input dan lapisan output sering disebut lapisan tersembunyi/ *hidden layer*). Jaringan dengan banyak lapisan ini dapat menyelesaikan permasalahan yang lebih sulit dari pada lapisan dengan lapisan tunggal, tentunya dengan pembelajaran yang lebih rumit.

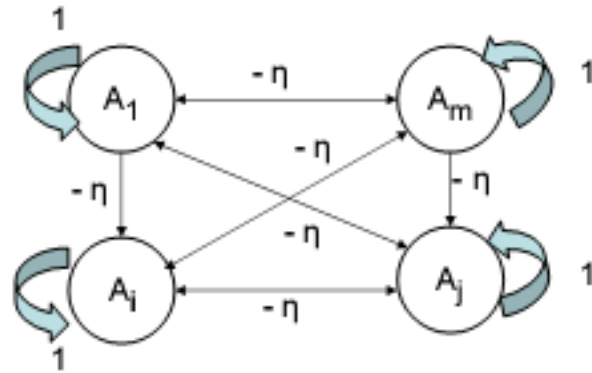


Gambar 5. Jaringan Syaraf Tiruan Dengan Lapisan Banyak

### 3. Jaringan dengan lapisan kompetitif (*competitif layer net*)

Hubungan antar-neuron pada lapisan kompetitif ini tidak diperlihatkan pada diagram arsitektur.

W<sub>32</sub>



Gambar 6. Jaringan dengan lapisan kompetitif