

## TEKNIK PERAMALAN TINGKAT PENJUALAN DENGAN JARINGAN SYARAF TIRUAN

Sri Mulyana

Program Studi Ilmu Komputer F MIPA UGM

Sekip Unit III Yogyakarta Telp (0274)546194

e-mail : [smulyana@ugm.ac.id](mailto:smulyana@ugm.ac.id)

### Abstrak

Tingkat penjualan merupakan salah satu tolok ukur keberhasilan dalam usaha perdagangan. Pada penelitian ini telah dikembangkan teknik peramalan tingkat penjualan berbasis jaringan syaraf tiruan (JST) dengan metode propagasi balik.

Tingkat penjualan berbasis JST yang dikembangkan terdiri 6 neuron masukan yang merupakan nilai parameter yang berpengaruh pada tingkat penjualan dan 1 neuron keluaran yang merupakan tingkat penjualan hasil prakiraan. Parameter-parameter tersebut meliputi : cuaca, tingkat hari libur (dalam 1 minggu), ada dan tidaknya event/kegiatan yang diselenggarakan di kota tempat restoran tersebut berada, banyaknya menu spesial yang ditawarkan, biaya iklan yang dikeluarkan dan frekuensi iklan di media. Adapun luaran sistem adalah tingkat penjualan yang dicapai restoran tersebut selama 1 minggu.

Dalam penelitian ini digunakan 90 data penjualan dengan 80 data digunakan untuk pelatihan dan 10 data digunakan untuk pengujian. Dari hasil pengujian terhadap 10 data penjualan, JST dengan metode propagasi balik diperoleh tingkat penyimpangan rata-rata sebesar 3.3 %

**Keywords :** Jaringan syaraf tiruan (JST), propagasi balik, tingkat penjualan

### 1. PENDAHULUAN

Penjualan merupakan salah satu tolok ukur keberhasilan dalam suatu usaha perdagangan. Banyak faktor yang mempengaruhi tinggi rendahnya tingkat penjualan diantaranya produk itu sendiri, harga, distribusi, promosi dan layanan purna jual. Prakiraan tingkat penjualan yang tepat dapat dijadikan rujukan guna menentukan keberlangsungan usaha dan tingkat keuntungan yang ingin dicapai.

Jaringan Syaraf Tiruan (JST), telah banyak digunakan untuk menganalisa data yang kompleks dan untuk mengenali pola. Hasilnya cukup menjanjikan untuk pengolahan data yang kompleks. Untuk itu telah dilakukan penelitian tentang teknik peramalan penjualan dengan menggunakan Jaringan Syaraf Tiruan (JST). Terdapat beberapa metode peramalan antara lain : metode pemulusan (*Smoothing*), metode dekomposisi, dan metode regresi [Makridakis, 1998].

Penelitian ini bertujuan membuat aplikasi jaringan syaraf tiruan yang dalam hal ini adalah jaringan syaraf tiruan *feed-forward* dengan algoritma pelatihan propagasi balik untuk peramalan tingkat penjualan. Dengan sistem ini dapat dimanfaatkan oleh para pengusaha untuk membuat prakiraan tingkat penjualannya berdasarkan pencapaian pada periode sebelumnya.

### 2. TINJAUAN PUSTAKA

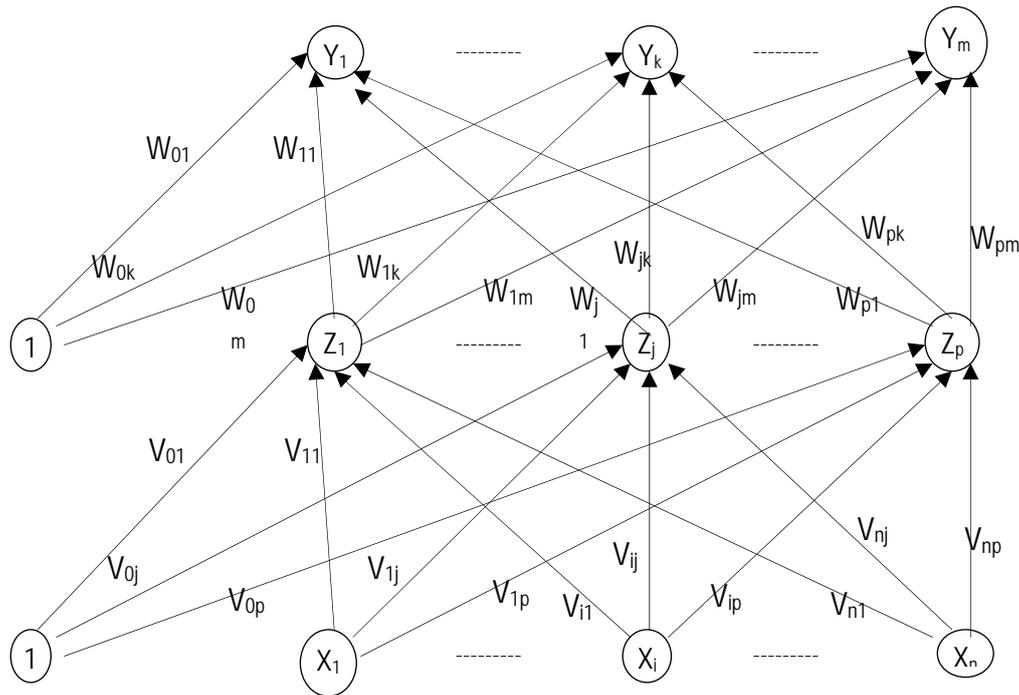
Jaringan Syaraf Tiruan (JST) telah digunakan dalam banyak implementasi yang melibatkan banyak data dan juga pada pengenalan pola. Beberapa riset telah dilakukan diantaranya adalah Jaringan Syaraf Tiruan (JST) untuk melakukan peramalan tingkat pertumbuhan PDB (Produk Domestik Bruto) [Tkaz, 1999].

Implementasi lain juga telah dilakukan pada penelitian penggunaan Jaringan Syaraf Tiruan (JST) untuk peramalan dan analisis data Pemasaran [Yau dkk, 1998]. Pada penelitian tersebut JST diaplikasikan pada data penjualan TV berwarna di Singapura yang dihimpun selama satu setengah tahun. Beberapa hal yang diindikasikan sebagai parameter yang dapat mempengaruhi tingkat penjualan adalah harga, ukuran layar, sistem stereo, faktor musim dan layar datar. Sedangkan implementasi JST dengan teknik propagasi balik juga telah dilakukan untuk perencanaan penjualan sistem komputer paralel [Thiesing,1999]. Pada penelitian ini telah dilakukan pembuatan peramalan penjualan dengan menggunakan Jaringan Syaraf Tiruan (JST).

Jaringan Syaraf Tiruan (JST) adalah sistem pemroses informasi yang memiliki karakteristik unjuk kerja tertentu yang menyerupai jaringan syaraf biologi (Fausett, 1994).

JST propagasi balik merupakan JST dengan jaringan *feed-forward* dengan banyak lapis yang dilatih menggunakan algoritma pelatihan propagasi balik. JST ini telah banyak dipakai dalam berbagai bidang seperti data mining, pembacaan tulisan tangan, simulasi kemudi truk, serta untuk pengenalan pola.

Sebuah JST lapis banyak dengan satu lapis unit tersembunyi (unit Z) ditunjukkan pada gambar-1 di bawah :



Gambar 1 : JST Propagasi balik dengan satu lapisan tersembunyi

Unit keluaran (unit Y) dan unit tersembunyi juga memiliki bias. Bias pada unit keluaran  $Y_k$  ditunjukkan dengan  $w_{0k}$  dan bias pada unit tersembunyi  $Z_j$  adalah  $v_{0j}$ . Bias berlaku seperti bobot pada hubungan dan memiliki nilai 1.

Pelatihan Jaringan Syaraf Tiruan (JST) dengan propagasi balik meliputi tiga tahap: tahap f, yaitu melatih pola masukan (dari unit masukan ke unit keluaran), *feed-forward* propagasi balik dan menghitung galat (dari unit keluaran ke unit masukan) dan tahap penyesuaian bobot. [Fausett, Laurene, 1994]

Selama tahap *feed-forward* (maju), setiap unit masukan ( $X_i$ ) menerima sinyal masukan dan mengirim sinyal ini ke setiap unit tersembunyi  $Z_1, \dots, Z_p$ . Setiap unit tersembunyi menghitung aktifasinya dan mengirim sinyalnya ( $z_j$ ) ke setiap unit keluaran. Setiap unit keluaran ( $Y_k$ ) menghitung aktifasinya ( $y_k$ ) untuk menunjukkan respon jaringan terhadap pola masukan yang diberikan.

Selama pelatihan, untuk setiap unit keluaran dibandingkan aktifasi  $y_k$  dengan targetnya  $t_k$  untuk menentukan galat antara pola masukan dengan unit keluaran tersebut. Setelah didapat galat, faktor  $\delta_k$  ( $k=1, \dots, m$ ) dihitung  $\delta_k$  yang digunakan untuk mendistribusikan galat pada unit keluaran  $Y_k$  kembali ke seluruh unit pada lapis sebelumnya (unit tersembunyi yang terhubung dengan  $Y_k$ ). Kemudian galat ini dipakai untuk mengubah bobot antara keluaran dengan lapisan tersembunyi. Dengan cara yang sama, faktor  $\delta_j$  ( $j = 1, \dots, p$ ) dihitung untuk setiap unit  $Z_j$ . Faktor  $\delta_j$  digunakan untuk mengubah bobot antara lapisan tersembunyi dengan lapisan masukan.

Setelah seluruh faktor  $\delta$  ditentukan, bobot untuk seluruh lapisan langsung disesuaikan. Penyesuaian bobot  $w_{jk}$  (dari unit tersembunyi  $Z_j$  ke unit keluaran  $Y_k$ ) didasarkan pada faktor  $\delta_k$  dan aktifasi dari unit  $Z_j$ , yaitu  $z_j$ . Penyesuaian bobot  $v_{ij}$  (dari unit masukan  $X_i$  ke unit tersembunyi  $Z_j$ ) adalah didasarkan pada faktor  $\delta_j$  dan aktifasi unit masukan  $x_i$ .

Fungsi aktifasi yang biasanya dipakai untuk melatih JST propagasi balik adalah fungsi sigmoid, baik biner maupun bipolar. Berikut algoritma pelatihannya (Fausett, Laurene, 1994) :

- Langkah 0.** Inisialisasi bobot (menentukan suatu nilai random kecil)
- Langkah 1.** Selama kondisi berhenti bernilai salah, dilakukan langkah 2-9 :
- Langkah 2.** Untuk setiap pasangan pelatihan, dilakukan langkah 3-8: (*Feedforward*)

**Langkah 3.** Setiap unit masukan ( $X_i, i = 1, \dots, n$ ) menerima sinyal masukan  $x_i$  dan mengirim sinyal ini ke seluruh unit pada lapisan berikutnya (lapisan tersembunyi).

**Langkah 4.** Untuk setiap unit tersembunyi ( $Z_j, j = 1, \dots, p$ ), sinyal masukan terboboti dijumlahkan  $\left( z_{in_j} = v_{0j} + \sum_{i=1}^n x_i v_{ij} \right)$  dan diterapkan fungsi aktivasi untuk menghitung keluarannya ( $z_j = f(z_{in_j})$ ) dan mengirim sinyal ini ke seluruh unit pada lapisan berikutnya (lapisan keluaran).

**Langkah 5.** Untuk setiap unit keluaran ( $Y_k, k = 1, \dots, m$ ) sinyal masukan terbobotnya dijumlahkan  $\left( y_{in_k} = w_{0k} + \sum_{j=1}^p z_j w_{jk} \right)$  dan diterapkan fungsi aktivasi untuk menghitung sinyal keluarannya ( $y_k = f(y_{in_k})$ )

**Propagasi balik dari galat :**

**Langkah 6.** Untuk setiap unit keluaran ( $Y_k, k = 1, \dots, m$ ) menerima sebuah pola target yang bersesuaian dengan pola masukan, dihitung galatnya ( $\delta_k = (t_k - y_k) f'(y_{in_k})$ ) dan dihitung koreksi bobotnya ( $\Delta w_{jk} = \alpha \delta_k z_j$ ) dan dihitung juga koreksi biasnya ( $\Delta w_{0k} = \alpha \delta_k$ ) dan mengirim  $\delta_k$  ke unit pada lapisan sebelumnya.

**Langkah 7.** Untuk setiap unit tersembunyi ( $Z_j, j = 1, \dots, p$ ), masukan deltanya dijumlahkan  $\left( \delta_{in_j} = \sum_{k=1}^m \delta_k w_{jk} \right)$  kemudian dikalikan dengan turunan fungsi aktifasinya untuk menghitung galatnya ( $\delta_j = \delta_{in_j} f'(z_{in_j})$ ), dihitung koreksi bobotnya ( $\Delta v_{ij} = \alpha \delta_j x_i$ ), dan koreksi biasnya ( $\Delta v_{0j} = \alpha \delta_j$ )

**Ubah bobot dan bias :**

**Langkah 8.** Untuk setiap unit keluaran ( $Y_k, k = 1, \dots, m$ ) bias dan bobotnya diubah ( $j = 1, \dots, p$ ) ( $w_{jk} (new) = w_{jk} (old) + \Delta w_{jk}$ ). Untuk setiap unit tersembunyi ( $Z_j, j = 1, \dots, p$ ) bobot dan biasnya diubah ( $i = 1, \dots, n$ ): ( $v_{ij} (new) = v_{ij} (old) + \Delta v_{ij}$ )

**Langkah 9.** Tes kondisi salah.

### 3. METODE PENELITIAN

Prosedur pelaksanaan penelitian ini meliputi pendalaman materi yang berkaitan dengan Jaringan Syaraf Tiruan (JST) baik dari buku-buku, paper, maupun dari internet. Tahap selanjutnya adalah merancang komponen Jaringan Syaraf Tiruan (JST) yang terdiri atas 3 layer, yaitu input, hidden layer dan output. Banyaknya neuron pada tiap layer adalah sebagai berikut, layer input 6 neuron yang merupakan parameter penentu tingkat penjualan, layer output hanya terdiri atas 1 neuron yaitu nilai tingkat penjualan yang dihasilkan, sedangkan pada hidden layer akan dilakukan uji coba sehingga diperoleh cacah neuron yang optimal. Dalam penelitian ini dipakai hidden layer sebanyak 6 buah neuron.

Tahap selanjutnya adalah mengumpulkan data penjualan beserta parameter yang mempengaruhi tingkat penjualan sebagai data untuk pelatihan/pembelajaran dalam Jaringan Syaraf Tiruan (JST). Pelatihan dilakukan pada sebagian data yang terkumpul, dan sisa data tingkat penjualan digunakan untuk membandingkan prakiraan oleh sistem dengan hasil yang sebenarnya. Tahap berikutnya adalah merancang dan mengimplementasikan program berbasis Jaringan Syaraf Tiruan (JST) untuk peramalan tingkat penjualan. Rancangan tersebut meliputi rancangan interface dan rancangan proses. Implementasi program tersebut menggunakan bahasa Borland Delphi 6.0 yang dioperasikan di bawah Windows XP. Dan yang terakhir adalah analisis hasil.

#### 4. HASIL DAN PEMBAHASAN

Sistem yang dibuat adalah sistem peramalan tingkat penjualan berbasis Jaringan Syaraf Tiruan (JST). Sistem ini digunakan untuk memperoleh prakiraan tingkat penjualan mingguan pada suatu restoran berdasarkan beberapa parameter.

Data yang digunakan untuk melatih sistem ini merupakan data sekunder sejumlah 90 data. Data tersebut berisi parameter-parameter yang mempengaruhi tingkat penjualan mingguan dan hasil penjualan yang diperoleh pada setiap minggunya. Parameter-parameter tersebut adalah : cuaca, tingkat hari libur (dalam 1 minggu), ada tidaknya event/kegiatan yang diselenggarakan di kota restoran tersebut berdomisili, banyaknya spesial menu yang ditawarkan, biaya iklan yang dikeluarkan dan banyaknya iklan di media. Keluaran sistem adalah tingkat penjualan yang dicapai restoran tersebut selama 1 minggu.

##### Persiapan Data

Data yang digunakan untuk pelatihan sistem di atas perlu dinormalkan agar pelatihannya baik dan memberikan kemudahan dalam pemrograman. Untuk parameter-parameter yang bernilai kategorial seperti : cuaca, tingkat hari libur (dalam 1 minggu), dan ada tidaknya event/kegiatan di kota tempat restoran tersebut, nilainya diubah ke bilangan antara 0 dan 1. Parameter yang mempengaruhi tingkat penjualan dinyatakan sebagai berikut :

Parameter uaca :

- Nilai 1 untuk kategori 'very high'
- Nilai 0,66 untuk kategori 'high'
- Nilai 0,33 untuk kategori 'medium'
- Nilai 0 untuk kategori 'low'

Parameter jumlah hari libur :

- Nilai 1 untuk kategori 'high'
- Nilai 0,5 untuk kategori 'medium'
- Nilai 0 untuk kategori 'low'

Parameter ada tidaknya event/kegiatan di kota restoran tersebut berdomisili :

- Nilai 1 untuk kategori 'yes'
- Nilai 0 untuk kategori 'no'

Untuk parameter-parameter yang bernilai numerik, seperti : biaya iklan dan tingkat penjualan, akan diubah ke bilangan antara 0 dan 1, yaitu dengan membagi dengan suatu bilangan skala. Untuk jumlah spesial menu yang ditawarkan, nilai pembagiannya adalah 5, biaya iklan nilai pembagiannya adalah 50000, jumlah iklan di media, nilai pembagiannya adalah 30, hasil penjualan, nilai baginya adalah 50000.

Pada saat pengujian, nilai yang dihasilkan oleh sistem, berupa perkiraan hasil penjualan akan dikalikan dengan nilai pembagiannya, yaitu 50000.

##### Pembahasan Sistem

Implementasi sistem dilakukan dengan data penjualan mingguan sebuah restoran sebanyak 90 data dan bisa ditambahkan setiap saat. Dari data tersebut, 80 data digunakan untuk pelatihan sedangkan sisanya untuk pengujian yaitu dengan membandingkan tingkat penjualan hasil peramalan dengan data tingkat penjualan sebenarnya yang dicapai.

Sistem peramalan penjualan ini menggunakan metode JST propagasi balik. Berdasarkan uji coba yang dilakukan, arsitektur JST yang dipilih 6-6-1, yaitu 6 neuron untuk lapisan masukan, 6 neuron untuk lapisan tersembunyi, dan 1 neuron untuk lapisan keluaran. Dengan menggunakan parameter belajar =1.

Fungsi aktifasi yang biasanya dipakai untuk melatih JST propagasi balik adalah fungsi sigmoid, baik yang biner maupun bipolar. Penjelasan algoritma yang di sampaikan di atas, dapat dijabarkan implementasinya sebagai berikut :

**Langkah 0.** Inisialisasi bobot (menentukan suatu nilai random kecil)

Proses tersebut diimplementasikan sebagai berikut :

{Inisialisasi bobot  $v$  (lapisan masukan ke lapisan tersembunyi)}

```
procedure bobot_awal_v;  
var i,j: byte;  
begin  
  for i:= 1 to 6 do  
    begin  
      for j:= 1 to 6 do  
        begin  
          v[i,j]:= random-0.5;  
        end;  
      end;  
    end;  
end;
```

{Inisialisasi bobot w (lapisan tersembunyi ke lapisan keluaran)}

```
procedure bobot_awal_w;  
var i: byte;  
begin  
  for i:=1 to 6 do  
    begin  
      w[i]:= random-0.5;  
    end;  
end;
```

**Langkah 1.** Selama kondisi berhenti adalah salah, dilakukan langkah 2-9 : yang diimplementasikan sebagai berikut :

{Kondisi salah jika indeks pengulangannya = 1000 (max epoch)}

```
for l:=1 to 1000 do  
  begin
```

**Langkah 2.** Untuk setiap pasangan pelatihan, dilakukan langkah 3-8:(*feed-forward*)

**Langkah 3.** Setiap unit masukan ( $X_i$ ,  $i = 1, \dots, n$ ) menerima sinyal masukan  $X_i$  dan mengirim sinyal ini ke seluruh unit pada lapisan berikutnya (lapisan tersembunyi).

**Langkah 4.** Untuk setiap unit tersembunyi ( $Z_j$ ,  $j = 1, \dots, p$ ) sinyal masukan terboboti dijumlahkan, fungsi aktifasi diterapkan untuk menghitung keluarannya, dan mengirim sinyal ini ke semua unit yang terdapat pada lapisan berikutnya (lapisan keluaran). Proses dari langkah-langkah tersebut adalah :

```
for k:=1 to 6 do  
  begin  
    z_in[k]:=0;  
    for i:=1 to 6 do  
      z_in[k]:= z_in[k] + (v[i,k]*input[i]);  
    z_in[k]:= z_in[k] + bias_input[k];  
    z[k]:= aktifasi(z_in[k]);  
  end;
```

**Langkah 5.** Untuk setiap unit keluaran ( $Y_k$ ,  $k=1, \dots, m$ ), sinyal masukan terbobotinya dijumlahkan, dan fungsi aktivasinya diterapkan untuk menghitung sinyal keluaran. Implementasi dari proses tersebut adalah :

```
for i:=1 to 6 do  
  begin  
    y_in:= y_in + (w[i] * z[i]);  
  end;  
  y_in:= y_in + bias_output;  
  y:= aktifasi(y_in);
```

**Langkah 6.** Untuk setiap unit keluaran ( $Y_k$ ,  $k=1, \dots, m$ ) menerima sebuah pola target yang bersesuaian dengan pola masukan, dihitung galatnya, yang direpresentasikan

```
delta_k:= (target[j]-y) * turunan(y_in);
```

menghitung koreksi bobotnyadankoreksi biasnyadan  $\delta_k$  dikirim ke unit pada lapisan sebelumnya, yang direpresentasikan sebagai berikut :

```
for i:=1 to 6 do
begin
    delta_w[i]:= alpha * delta_k * z[i];
    { w[i]:= w[i] + delta_w[i];}
end;
delta_bias_output:= alpha * delta_k;
{ bias_output:= bias_output + delta_bias_output;}
```

**Langkah 7.** Untuk setiap unit tersembunyi ( $Z_j, j=1, \dots, p$ ) masukan deltanya dijumlahkan, kemudian dikalikan dengan turunan fungsi aktifasinya untuk menghitung galatnya. Representasi dari proses tersebut adalah :

```
for i:=1 to 6 do
begin
    delta_in_j[i]:= delta_k * w[i];
    delta_j[i]:= delta_in_j[i] * turunan(z_in[i]);
end;
```

dihitung koreksi bobotnya, dan dihitung koreksi biasnya (digunakan untuk mengubah  $v_{0j}$ ) (mengubah bobot dan biasa). Representasi dari proses tersebut adalah :

```
for k:=1 to 6 do
begin
    for i:=1 to 6 do
    begin
        delta_v[i,k]:= alpha * delta_j[k] * input[i];
        v[i,k]:= v[i,k] + delta_v[i,k];
    end;
    delta_bias_input[k]:= alpha * delta_j[k];
    bias_input[k]:=bias_input[k]+delta_bias_input[k];
end;
```

**Langkah 8.** Untuk setiap unit keluaran ( $Y_k, k=1, \dots, m$ ) bias dan bobot ( $j = 1, \dots, p$ ) diubah dengan proses sebagai berikut :

```
for i:=1 to 6 do
begin
    delta_w[i]:= alpha * delta_k * z[i];
    w[i]:= w[i] + delta_w[i];
end;
delta_bias_output:= alpha * delta_k;
bias_output:= bias_output + delta_bias_output;
```

Untuk setiap unit tersembunyi ( $Z_j, j=1, \dots, p$ ) bobot dan biasnya ( $i=1, \dots, n$ ) diubah. Representasi dari proses tersebut adalah :

```
for k:=1 to 6 do
begin
    for i:=1 to 6 do
    begin
        delta_v[i,k]:= alpha * delta_j[k] * input[i];
        v[i,k]:= v[i,k] + delta_v[i,k];
    end;
    delta_bias_input[k]:= alpha * delta_j[k];
    bias_input[k]:=bias_input[k]+delta_bias_input[k];
end;
```

**Langkah 9.** Melakukan tes kondisi salah.

### Pengujian Sistem

Pengujian sistem dilakukan menggunakan 10 data dari penjualan pada minggu ke 81 hingga 90. Pengujian dilakukan dengan cara melatih sistem dengan data sebelumnya hingga minggu terakhir sebelum periode yang akan diperkirakan tingkat penjualannya. Sebagai ilustrasi dapat dijelaskan sebagai berikut : misalkan dilakukan pengujian untuk perkiraan tingkat penjualan pada minggu ke-84, maka telah dilakukan pelatihan pada sistem dengan menggunakan data tingkat penjualan real mulai minggu ke 81 hingga minggu ke-83.

Berikut ini contoh tampilan pengujian sistem pada minggu ke-90 :

Yang pertama dipilih menu **Tambah Data**, seperti pada gambar 2, sebagai berikut :



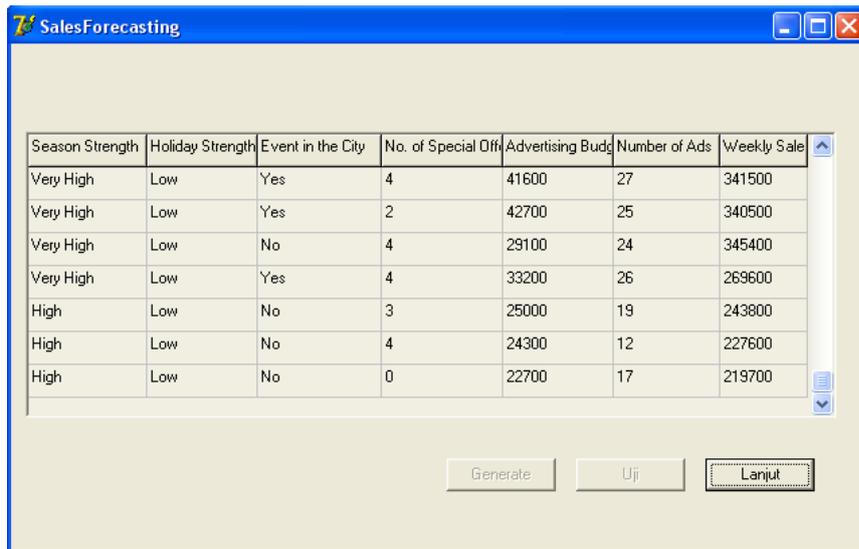
Gambar 2 : Form pilih menu

Pengisian data untuk penambahan data yang akan dilatih dengan data tingkat penjualan pada minggu ke-89, seperti terlihat pada gambar 3 berikut :



Gambar 3: Tampilan proses penambahan data

Untuk melihat data yang akan dilatih dengan memilih tombol **generate** dan selanjutnya sistem dapat melakukan pelatihan dengan memilih tombol **uji**, seperti pada gambar 4 berikut :



Season Strength	Holiday Strength	Event in the City	No. of Special Off	Advertising Budg	Number of Ads	Weekly Sale
Very High	Low	Yes	4	41600	27	341500
Very High	Low	Yes	2	42700	25	340500
Very High	Low	No	4	29100	24	345400
Very High	Low	Yes	4	33200	26	269600
High	Low	No	3	25000	19	243800
High	Low	No	4	24300	12	227600
High	Low	No	0	22700	17	219700

Gambar 4 : Tampilan data yang dilatih dan pelatihan

Setelah dilakukan proses pelatihan, pengujian terhadap prediksi tingkat penjualan pada minggu ke-90 dapat dilakukan dengan mengisi nilai parameter-parameter pada form pengujian dan memilih tombol **proses**, seperti yang terlihat pada gambar 5 berikut :



**PENGUJIAN**

Pengaruh cuaca : Medium  
Banyaknya Menu Spesial : 2  
Range 0-5

Banyaknya Hari Libur : None  
Biaya Iklan : 19800  
Range < 50000

Adanya Event di Kota : No  
Jumlah Iklan di Media : 14  
Range 1-30

Proses

Hasil Perkiraan Penjualan : 220210

Reset Keluar Menu

Gambar 5: Tampilan hasil pengujian

Data yang digunakan untuk proses pengujian adalah 10 buah yang merupakan capian tingkat penjualan pada minggu ke-81 sampai dengan minggu ke-90, yang selengkapnya dapat dilihat pada tabel berikut :

**Tabel 3.1 : Data untuk pengujian**

Week No.	Inputs						Target
	Season Strength	Holiday Strength	Event in the City	No. of Special Offerings	Advertising Budget	Number of Ads	Weekly Sales
81	High	None	No	0	22600	19	349200
82	Very High	None	No	4	39400	25	348200
83	Very High	None	Yes	4	41600	27	341500
84	Very High	None	Yes	2	42700	25	340500
85	Very High	None	No	4	29100	24	345400
86	Very High	None	Yes	4	33200	26	269600
87	High	None	No	3	25000	19	243800
88	High	None	No	4	24300	12	227600
89	High	None	No	0	22700	17	219700
90	Medium	None	No	2	19800	14	224800

Hasil pengujian terhadap data di atas menggunakan sistem prakiraan penjualan dengan Jaringan Syaraf Tiruan (JST) yang dikembangkan, memberikan hasil sebagai berikut :

**Tabel 3.2 : Data hasil pengujian**

Minggu ke	Perkiraan tingkat penjualan	Tingkat penjualan sebenarnya	selisih	Selisih dalam %
81	267236	349200	81964	23,47194
82	296302	348200	51898	14,90465
83	271262	341500	70238	20,5675
84	287312	340500	53188	15,62056
85	303572	345400	41828	12,11002
86	286226	269600	-16626	-6,166914
87	268071	243800	-24271	-9,955291
88	252744	227600	-25144	-11,04745
89	273675	219700	-53975	-24,56759
90	220210	224800	4590	-2,041815

Dari hasil tersebut dapat diketahui bahwa rata-rata selisih prakiraan menggunakan Jaringan Syaraf Tiruan (JST) yang dikembangkan sebesar 3,3 %

## 5. KESIMPULAN

Dari penelitian ini dapat disimpulkan sebagai berikut

- Jaringan syaraf tiruan telah dapat digunakan sebagai aplikasi peramalan tingkat penjualan.
- Tinglat Layer yang digunakan pada penelitian ini adalah 6-6-1 yaitu terdiri dari 6 neuron masukan, 6 neuron sebagai hidden layer dan 1 neuron keluaran.
- Dari hasil uji coba sistem yang dikembangkan diperoleh tingkat penyimpangan rata-rata sebesar 3,3 %

## 6. DAFTAR PUSTAKA

- Fausett, L., 1994, *Fundamental of Neural Networks*, Prentice Hall, Inc., New Jersey
- Makridakis, S., Wheelwright, S.C., McGee, V.E., 1998, *Forecasting : Methods and Applications*, Second edition, John Wiley & Sons, Inc.
- Thiesing, F.M., Middelberg U., Vornberger O.,1999, *Parallel Back-Propagation for Sales Prediction System*
- Tkacz, G., Sarah H., 1999, *Forecasting GDP Growth Using Artificial Neural Network*, Working Paper 99-3
- Yau J., Teng N., Poh H.L., Tan C.L., 1998, *Forecasting and Analysis of Marketing Data Using Neural Networks*, Journal of Information Science and Engineering 14, 843-862.