

BAB II

LANDASAN TEORI

2.1 Grafologi

Grafologi merupakan satu cabang ilmu untuk membaca karakter. Manusia pada umumnya tertarik dengan keanekaragaman dan keunikan individu. Dengan adanya grafologi seorang individu dapat berfokus dengan penafsiran karakter dan ciri kepribadian seseorang dengan menganalisa tulisan tangan seseorang tersebut. Kita bisa mengandalkan grafologi untuk mengetahui karakter dari seseorang.

2.1.1 Pengertian Grafologi

Secara bahasa, kata grafologi berasal dari bahasa Yunani, yaitu “grapho” yang berarti menulis, dan “logos” yang berarti sebuah cabang studi ilmiah. Dengan demikian grafologi dapat diartikan menjadi suatu ilmu yang dapat mengidentifikasi, mengevaluasi dan pemahaman akan karakter atau watak seseorang melalui pola tulisan tangannya (Dwi Sunar Prasetyo, 2010 dikutip oleh Nur Karimah, 2012).

Tulisan tangan seseorang merupakan hasil dari pikiran sadar dan bawah sadar dari seseorang tersebut yang mencerminkan dan menggambarkan bagaimana karakternya. Tulisan tangan memiliki informasi yang dapat mengidentifikasi tingkat kesehatan seseorang perkembangan jiwa dan sifat seseorang. Menurut beberapa ahli, ilmu grafologi ini memiliki ketepatan akurasi dalam mengidentifikasi karakter seseorang mendekati kesempurnaan. Akurasi dalam penggunaan grafologi ini mencapai 80% hingga 90% kesesuaian antara kondisi kejiwaan dan kepribadian seseorang dengan tulisan tangan seseorang tersebut (Siswanto dkk, 2011 dikutip oleh Fikri, 2012).

Proses – proses psikis maupun otak seseorang mempengaruhi pada gerakan psikomotorik menulis seseorang. Setiap orang memiliki khas tulisan tangan masing – masing karena terdapat rangsangan – rangsangan kecil pada otak seseorang tersebut mempengaruhi pola. Oleh karena itu, Beberapa faktor dari tulisan tangan seseorang datang dari pembawaan aslinya. Dengan demikian,

tulisan tangan itu bukan hasil karya otot tangan semata, melainkan gabungan dari gerakan-gerakan otak dan pembawaan psikis seseorang.

2.1.2 Sejarah Perkembangan Grafologi

Sejarah grafologi sudah ada sejak 6000 tahun yang lalu dari beberapa sumber menyebutkan bahwa ilmu grafologi sudah ada di Cina. Para ilmuwan sudah menerapkan coretan tangan untuk mengetahui karakteristik seseorang pada waktu itu. Seorang filsuf dari Italia dr. Camillo Baldi pada tahun 1622 dan juga guru besar dari Universitas Bologna dalam bukunya *A Method to Recognize the Nature and Quality of a Writer* mengemukakan penemuan tentang ilmu pengenalan tulisan yang dibuat secara sistematis dan ilmiah. Buku ini menjadi buku pertama yang menganalisis tentang tulisan tangan. Kemudian Pastor Swiss (J.C Lavanter) pada tahun 1741-1801, membuat laporan yang lebih sistematis dan cermat tentang tulisan tangan.

Seorang ilmuwan Perancis Abbe Jean Hyppolyte Michon pada abad ke-19, membentuk *Graphological Society* (Lembaga Grafologi) di Paris, yang berkembang sampai masa Perang Dunia Kedua (1939-1945). Berbekal ketajaman pengamatan yang tinggi dan pengalaman, Michon berhasil menghimpun katalog yang penuh dengan ciri-ciri keteraturannya dan tanda-tanda grafis. Maka pada saat itu Michon dianggap sebagai Bapak Grafologi modern hingga sekarang karena telah berhasil menerapkan dasar-dasar Grafologi Modern.

Di Indonesia sendiri, Grafologi belum berkembang secara optimal. Hal ini disebabkan karena belum adanya ahli-ahli grafologi yang dapat membawa ilmu ini di tengah-tengah masyarakat luas. Perkembangan Grafologi di Indonesia umumnya hanya ditempatkan sebagai sebuah teori dalam mata kuliah psikologi dan sebagai alat bantu Psikodiagnostik atau Tes Psikologi di kota-kota besar.

2.1.3 Penerapan Grafologi

Meskipun kebutuhan manusia saat ini banyak yang sudah dialihkan pada teknologi, Namun grafologi dianggap masih sangat berguna. Apalagi ketepatan menilai tulisan tangan sangatlah membantu dalam banyak bidang saat ini.

Contohnya pada bidang pendidikan, Pada bidang pendidikan kita dapat mengetahui bakat dan minat seorang siswa, perilaku kekerasan disekolah, dan bisa juga digunakan untuk bimbingan konseling. Grafologi dapat digunakan dibidang kriminal, forensik, dan pengembangan kepribadian seseorang. Grafologi dapat digunakan oleh para ahli untuk mendiagnosa penyakit keterbelakangan mental dan dapat digunakan oleh polisi untuk mendapatkan gambaran tentang kesehatan mental tersangka. Dengan manfaat grafologi ini, seseorang bisa juga membaca kepribadian calon pegawai atau bawahan dalam sebuah institusi.

2.1.4 Teknik Grafologi

Adapun penelitian tentang grafologi banyak dilakukan seperti parameter yang paling mencolok dari tulisan tangan yang digunakan dalam analisis grafologi antara lain bentuk garis dasar tulisan, pengenalan huruf-huruf alphabet, margin tulisan, spasi baris, garis miring, kemiringan kata, ketajaman sudut, ukuran huruf, kerapatan teks, kecepatan menulis dan keteraturan menulis. Pada penelitian ini parameter yang akan peneliti gunakan adalah huruf t kecil pada tulisan tangan.

2.1.5 Grafologi Huruf t

Dalam Banyak hal garis bar pada huruf t menunjukkan seberapa tinggi atau rendahnya cara paling cepat untuk memprediksi keinginan seseorang. Huruf t bisa menggambarkan pandangan seseorang terhadap suatu profesi terhadap gambaran diri yang ditampilkan. Proses analisis dilakukan pada garis bar huruf t yang menggambarkan sikap seorang pada suatu waktu terhadap pemikiran karir dan juga harapan.

2.1.5.1 Huruf t Dengan Garis Bar Menaik

Menggambarkan seseorang yang mempunyai pikiran optimis, dinamis, dan mempunyai motivasi lebih. Selain itu mendeskripsikan seseorang yang mempunyai semangat, kepercayaan diri, dan energik.



Gambar 2.1 Huruf t dengan garis bar menaik

2.1.5.2 Huruf t Dengan Garis Bar Normal

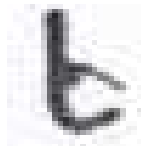
Menggambarkan seseorang yang tenang, dapat mengendalikan diri dalam mengambil keputusan dan melakukan tindakan.



Gambar 2.2 Huruf t dengan garis bar normal

2.1.5.3 Huruf t Dengan Garis Bar Menurun

Menunjukkan individu yang suka bergantung pada orang lain, mudah putus asa dan pasrah, serta kurang termotivasi.



Gambar 2.3 Huruf t dengan garis Bar menurun

2.2 Citra Digital

Citra digital adalah sebuah array yang berisikan nilai asli maupun kompleks yang direpresentasikan menggunakan deret bit tertentu. Suatu citra bisa diartikan menjadi fungsi $f(x,y)$ yang berukuran M baris dan N kolom, dengan x dan y merupakan koordinat spasial, dan amplitude f di titik koordinat (x,y) dinamakan intensitas dari citra pada titik tersebut. Jika nilai x , y dan nilai amplitude f secara merata berhingga (finite) dan mempunyai nilai diskrit maka bisa dikatakan bahwa citra tersebut merupakan citra digital.

Adapun tiga jenis citra yang bisa diolah dalam pemrosesan citra. Ketiga jenis citra tersebut adalah citra berwarna (RGB), citra berskala keabuan (*grayscale*), dan citra biner (*monochrome*).

2.2.1 Citra Warna (RGB)

Citra warna atau yang disebut dengan citra RGB adalah suatu model warna yang terdiri dari merah, hijau, biru yang digabungkan dalam memberntuk

suatu susunan warna yang luas. Kedalaman piksel citra yang digunakan mempengaruhi banyaknya warna yang digunakan. Untuk layar komputer, nilai rentang paling minim adalah 0 sedangkan yang paling maksimum adalah 255. Pilihan skala 256 ini diacukan pada cara mengungkap 8 digit bilangan biner yang dipakai oleh komputer. Menggunakan cara ini, akan dihasilkan warna gabungan sejumlah $256 \times 256 \times 256 = 1677726$ macam warna. Satu jenis warna, dapat diilustrasikan menjadi sebuah vector di ruang 3 dimensi yang umumnya digunakan dalam matematika, koordinat dinyatakan kedalam bentuk tiga bilangan, yaitu komponen x, komponen y, dan komponen z.

2.2.2 Citra Berskala Keabuan (*Grayscale*)

Pada citra grayscale, setiap pikselnya mempunyai nilai lain diantara hitam dan putih yang biasa disebut nilai abu-abu. Banyaknya kemungkinan warna abu-abu yang muncul dipengaruhi oleh bit citra tersebut. Contohnya suatu citra memiliki kedalaman sebesar 8 bit. Jadi citra tersebut memiliki skala warna sejumlah 256 atau 256 skala yaitu skala nol sampai 255 (0-255). Angka 0 mewakili warna hitam dan angka 255 mewakili warna putih dan lainnya mewakili warna abu-abu. Warna abu-abu skala 200 memiliki warna yang lebih kegelapan jika dibandingkan dengan warna abu-abu skala 201.

2.2.3 Citra Biner (Monokrom)

Citra biner atau citra monokrom adalah citra *digital* yang cuma memiliki dua pilihan nilai *pixel* yaitu hitam dan putih. Citra biner dapat disebut sebagai citra *black and white*. Hanya dibutuhkan 1 *bit* untuk mewakili nilai setiap *pixel* dari citra biner.

2.3 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) merupakan salah satu metode yang digunakan untuk mengidentifikasi pola dalam data dan mengekspresikannya sedemikian rupa. PCA juga merupakan sebuah transformasi linear yang sering digunakan untuk klasifikasi dan kompresi data. Cara kerja metode PCA adalah

dengan menghilangkan korelasi di antara variabel bebas dengan cara mentransformasikan variabel bebas awal ke variabel bebas baru yang tidak memiliki korelasi sama sekali. Variabel baru ini berisi nilai-nilai komponen utama atau principal component (PC) dari variabel tersebut.

Tahapan PCA pada dasarnya bertujuan untuk menyederhanakan bilangan yang diamati dengan cara mereduksi dimensinya (Abdi, 2010). Itulah salah satu kelebihan dari metode PCA sehingga dapat mengurangi waktu pengolahan datanya. Dalam perhitungannya, metode PCA melibatkan nilai eigen dari matriks kovarian. PCA juga dikenal dengan sebutan KLH (Karhunen-Loeve Transformation). PCA juga sering disebut dengan metode eigenface yang banyak digunakan untuk pengenalan citra wajah.

2.3.1 Algoritma Principal Component Analysis (PCA)

Dalam penggunaan metode PCA, ada beberapa hal yang harus diperhatikan yaitu citra harus tepat berada ditengah dan memiliki ukuran yang sama dengan data latih. Berikut ini adalah langkah-langkah dalam proses PCA atau *eigenface* (Turk, 1991) :

1. Membuat matriks data set citra huruf t

Langkah pertama dalam metode PCA atau *eigenface* adalah menyiapkan data dengan membuat suatu himpunan S yang terdiri dari seluruh training image. Setiap citra diubah ke dalam sebuah vektor berukuran n yang disebut dengan vektor citra data set (data latih) dan ditempatkan ke dalam himpunan S. Matriks data set S adalah matriks berukuran N x n yang setiap barisnya berisi vektor- vektor citra r_i . Dimana N adalah jumlah citra. Representasi matriks s dapat digambarkan sebagai matriks berikut :

$$S = \begin{bmatrix} r1 \\ r2 \\ r3 \\ \vdots \\ rN \end{bmatrix} = \begin{bmatrix} r11 & r12 & r13 & \dots & r1n \\ r21 & r22 & r23 & \dots & r2n \\ r31 & r32 & r33 & \dots & r3n \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ rN1 & rN2 & rN3 & \dots & rNn \end{bmatrix}$$

$$S = \{r_1, r_2, r_3, \dots, r_N\}$$

2. Membuat matriks rata-rata (*mean*)

Rumus perhitungan matriks rata-rata adalah sebagai berikut :

$$\bar{r} = \frac{1}{M} \sum_{n=1}^M r_n = \frac{1}{M} T_{11}$$

Dari rumus diatas maka akan didapatkan hasil $\bar{r} = [\bar{r}_1, \bar{r}_2, \bar{r}_3 \dots \bar{r}_m]$

\bar{r} = matriks rata-rata (*mean*)

M = banyaknya data di data set

r_n = data ke n

3. Membuat matriks selisih

Matriks selisih dibuat dengan mengurangi setiap elemen matriks pada kolom ke-i dengan \bar{r}_i . Representasi matriks dapat dilihat pada rumus dibawah :

$$r_{ij} - \bar{r}_i$$

Dari rumus diatas dapat direpresentasikan kedalam matriks selisih berikut ini :

$$= \begin{bmatrix} r_{11} - \bar{r}_1 & r_{12} - \bar{r}_2 & r_{13} - \bar{r}_3 & \dots & r_{1n} - \bar{r}_n \\ r_{21} - \bar{r}_1 & r_{22} - \bar{r}_2 & r_{23} - \bar{r}_3 & \dots & r_{2n} - \bar{r}_n \\ r_{31} - \bar{r}_1 & r_{32} - \bar{r}_2 & r_{33} - \bar{r}_3 & \dots & r_{3n} - \bar{r}_n \\ \dots & \dots & \dots & \dots & \dots \\ r_{N1} - \bar{r}_1 & r_{N2} - \bar{r}_2 & r_{N3} - \bar{r}_3 & \dots & r_{Nn} - \bar{r}_n \end{bmatrix}$$

= matriks selisih

r_{ij} = data ke-i

\bar{r}_i = data rata-rata (*mean*)

4. Membuat matriks kovarian C.

Rumus perhitungan matriks kovarian C adalah sebagai berikut :

$$C = \frac{1}{M} \sum_{n=1}^M (\Phi_n)(\Phi_n)^t = A^t a \quad L = A^t A$$

$$A = \{ \Phi_1 \Phi_2 \Phi_3 \dots, \Phi_n \}$$

C = matriks kovarian berdimensi n x n

A = berisi matriks selisih

= jumlah data set

5. Menghitung eigenvektor (v) dan eigenvalue (λ) dari matriks kovarian (C) Rumus untuk menghitung eigenvektor dan eigenvalue dari matriks kovarian adalah sebagai berikut:

$$Cv = \lambda v$$

Dari rumus diatas, maka didapatkan sebuah persamaan :

$$(Cv - \lambda v) = 0$$

$$(C - \lambda I)v = 0 \text{ atau } \det(\lambda I - C) = 0$$

Persamaan diatas digunakan untuk mencari nilai *eigenvalue* (λ), sedangkan untuk mencari nilai *eigenvector* (v), maka kita akan mensubtitusikan nilai dari *eigenvalue* (λ) ke dalam persamaan :

$$(\lambda I - C)v = 0$$

Keterangan :

λ = *eigenvalue* atau nilai eigen

v = *eigenvector* atau vektor eigen

C = matriks kovarian

I = matriks identitas

6. Mencari nilai *principal component* (PC).

Pada proses 5 akan dihasilkan n *eigenvector* dan n *eigenvalue*.

Dari n *eigenvector* diambil m *eigenvector* v yang memiliki *eigenvalue* terbesar. Karena *eigenvector* ini memiliki dimensi yang sama dengan citra awal, maka vektor eigen ini juga bisa disebut *principal component* (PC) atau komponen utama. Nilai PC dapat juga dicari dengan rumus sebagai berikut :

$$u_l = \sum_{k=1}^M v_{lk} \Phi_k$$

$l = 1, \dots, M$

Keterangan :

u = nilai *principal component* (PC)

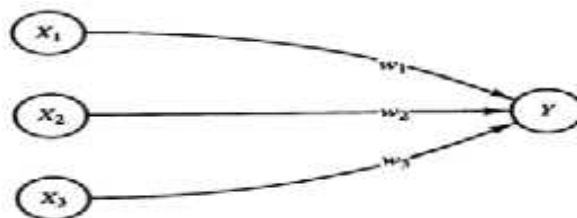
v = *eigenvector*

= nilai selisih

M = jumlah data

2.4 Jaringan Syaraf Tiruan (JST)

Jaringan Syaraf Tiruan (JST) adalah sistem pengolahan informasi yang memiliki kemampuan belajar, mengingat dan menyelesaikan suatu masalah berdasarkan proses belajar yang sudah diberikan dan mengambil keputusan dengan menirukan cara kerja otak manusia (pakar atau ahli). Jaringan Syaraf Tiruan bisa mengenali kegiatan dengan berbasis pada data masa lalu. Data masa lalu akan dipelajari oleh jaringan sayraf tiruan sehingga mempunyai kemampuan untuk mnghasilkan keputusan terhadap data yang belum pernah dipelajari. Sistem ini mempunyai karakteristik yang meniru jaringan syaraf biologi yang berupa hubungan antar *neuron* (arsitektur), metode penentuan bobot pada saluran penghubung (*learning algorithm*) dan fungsi aktivasi yang digunakan.



Gambar 2.4 Gambaran JST

Pada gambar 2.1, Secara gambaran jaringan syaraf tiruan terdiri dari tiga *neuron* pada *layer input* dan satu *neuron* pada *layer output*. *Neuron Y* berfungsi menerima *input* dari neuron x_1 , x_2 , x_3 . Nilai aktivasi neuron tersebut yaitu x_1 , x_2 , x_3 . Bobot saluran penghubung dari x_1 , x_2 , x_3 ke neuron Y adalah w_1 , w_2 , w_3 . *Input* jaringan, y_{in} ke neuron Y adalah jumlah dari bobot sinyal x dari neuron-neuron x_1 , x_2 , x_3 .

$$Y_{in} = w_1x_1 + w_2x_2 + w_3x_3$$

Neuron pada jaringan syaraf tiruan disusun kedalam suatu *layer* yang membentuk suatu arsitektur jaringan syaraf tiruan. Biasanya *layer* tersebut terdiri dari *layer input*, *hidden layer* dan *layer output*.

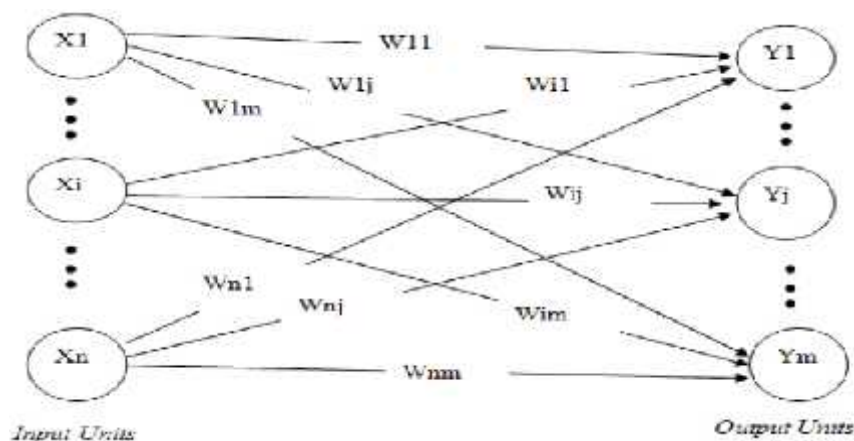
Jaringan syaraf tiruan mempunyai dua jenis metode penentuan bobot yaitu :

- a. *Supervised training* adalah metode penentuan bobot yang menerapkan sepasang kumpulan vektor yaitu vektor pelatihan dan vektor target. Penentuan bobot diacukan pada perbandingan antara vektor pelatihan dan target sampai *output* jaringan syaraf tiruan sama dengan targetnya.
- b. *Unsupervised training* adalah *self-organizing* jaringan syaraf tiruan, maksudnya menggunakan vektor pelatihan tanpa vektor target. Jaringan syaraf memodifikasi bobot sehingga vektor-vektor *input* yang sama dikelompokkan ke suatu unit *output* yang sama (*cluster*) dan konsisten. Dengan demikian penerapan salah satu vektor pelatihan atau suatu vektor yang serupa akan menghasilkan *output* yang sama.

2.4.1 Arsitektur Jaringan

2.4.1.1 Single Layer

Arsitektur jaringan *single layer* merupakan sekumpulan *input* neuron dihubungkan langsung dengan sekumpulan *output*nya .

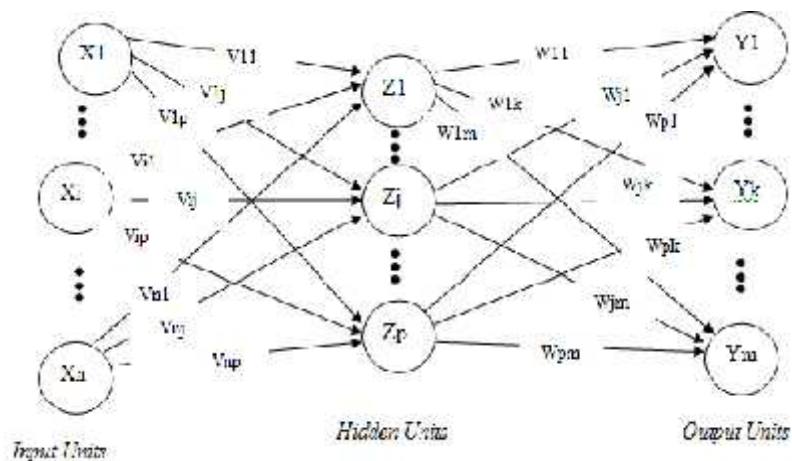


Gambar 2.5 Arsitektur Jaringan Single Layer

Pada gambar di atas dapat dilihat jika arsitektur jaringan *single layer* dengan n buah masukan (X_1, X_2, \dots, X_n) dan m buah keluaran (Y_1, Y_2, \dots, Y_m). Pada jaringan ini semua unit *input* dihubungkan dengan semua unit *output*.

2.4.1.2 Multi Layer

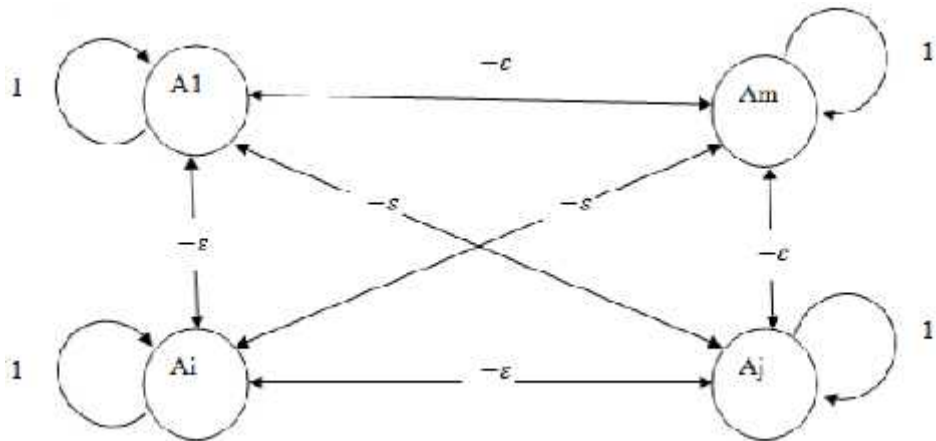
Arsitektur jaringan *multi layer* merupakan jaringan dengan n buah unit masukan (X_1, X_2, \dots, X_n), satu layar tersembunyi yang terdiri dari m buah unit (Z_1, Z_2, \dots, Z_m) dan m buah unit keluaran (Y_1, Y_2, \dots, Y_m). Jaringan *multi layer* bisa menyelesaikan masalah yang lebih rumit jika dibandingkan dengan *single layer*, meskipun kadang kala proses pelatihan lebih rumit dan lama.



Gambar 2.6 Arsitektur Jaringan *Multi Layer*

2.4.1.3 Competitive Layer

Model arsitektur jaringan *competitive layer* mirip dengan *single layer* ataupun *multi layer*. Perbedaannya, ada *neuron output* yang memberikan sinyal pada unit *input* sering disebut *feedback loop*. Dengan demikian sinyal mengalir secara dua arah, yaitu maju dan mundur.



Gambar 2.7 Arsitektur Jaringan *Competitive Layer*

2.4.2 Pemodelan Jaringan Syaraf Tiruan

Tahapan pemodelan dengan ANN secara umum terdiri dari empat langkah yaitu:

1. Menyiapkan data
2. Memilih arsitektur
3. Pembelajaran (*learning*) dan pengujian (*testing*)

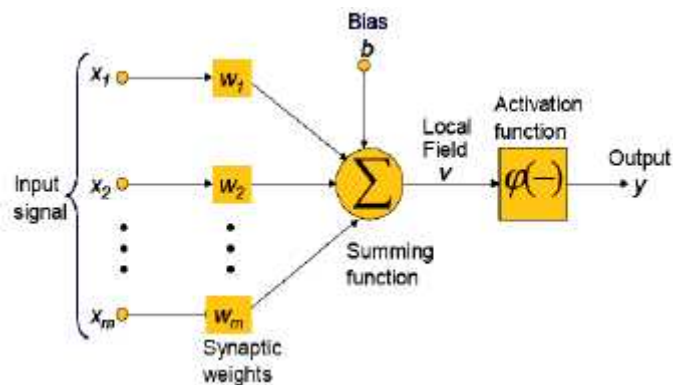
Untuk menilai kecocokan dengan menggunakan fungsi objektif atau fungsi kesalahan yang merupakan persamaan dari perhitungan dan pengamatan. Bentuk dari fungsi objektif digunakan kesalahan absolute rata-rata (KAR). Persamaan yang diformulasikan (Hadihardaja 2005) :

$$\frac{1}{n} = \frac{\sum (Q_{omp} - Q_{obs})}{Q}$$

Q_{omp} = debit hasil perhitungan

Q_{obs} = debit hasil pengamatan

N = Banyak data



Gambar 2.8 Model Jaringan Syaraf Tiruan

Kunggulan – keunggulan yang ada pada JST yaitu :

1. Belajar Adaptive : mampu untuk belajar bagaimana melakukan pekerjaan berdasarkan data yang diberikan untuk pelatihan.
2. *Self-Organisation*. Jaringan syaraf tiruan bisa membuat kelompok sendiri dari informasi yang diterima ketika waktu pembelajaran.
3. *Real Time Operation*. Perhitungan jaringan syaraf tiruan dapat dilakukan secara parallel sehingga perangkat keras yang dirancang secara khusus dapat mengambil keuntungan dari kemampuan ini.

Selain memiliki keunggulan - keunggulan tersebut, JST memiliki kelemahan-kelemahan sebagai berikut.

1. Tidak efektif jika digunakan untuk mengerjakan operasi-operasi numerik dengan presisi tinggi.
2. Tidak efisien jika digunakan untuk mengerjakan operasi algoritma aritmatik, dan simbolis.
3. Untuk operasi JST sangat butuh pelatihan sehingga bila jumlah datanya banyak, waktu yang dipakais untuk operasi proses sangat lama.

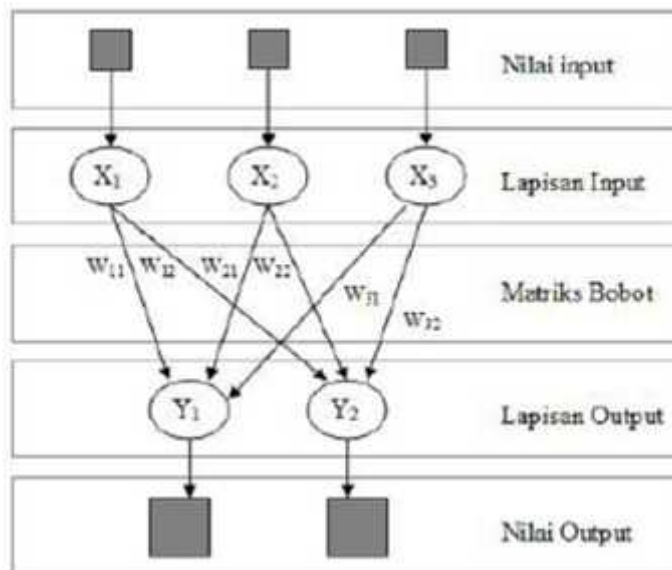
2.5 Learning Vector Quantization

LVQ adalah metode pengenalan pola yang mengerjakan pembelajaran terlebih dahulu, Kusumadewi (2003). LVQ merupakan metode klasifikasi pola

yang tiap -tiap unit keluaran mewakili kategori (sejumlah unit keluaran biasanya digunakan untuk tiap - tiap kelas). Vektor nilai untuk sebuah unit keluaran umumnya dinyatakan menjadi sebuah vektor referens. Diperkirakan jika susunan pola pelatihan dengan klasifikasi yang tersedia dengan distribusi awal vektor referens. Setelah pelatihan, jaringan LVQ mengelompokkan vektor masukan dengan menugaskan ke kelas yang sesuai sebagai unit keluaran, sedangkan yang mempunyai vektor referens dikelompokkan sebagai vektor masukan.

2.5.1 Arsitektur Jaringan

LVQ adalah jaringan syaraf dengan tipe arsitektur jaringan lapis tunggal umpan-maju (Single Layer Feedforward) yang terdiri atas unit masukan.

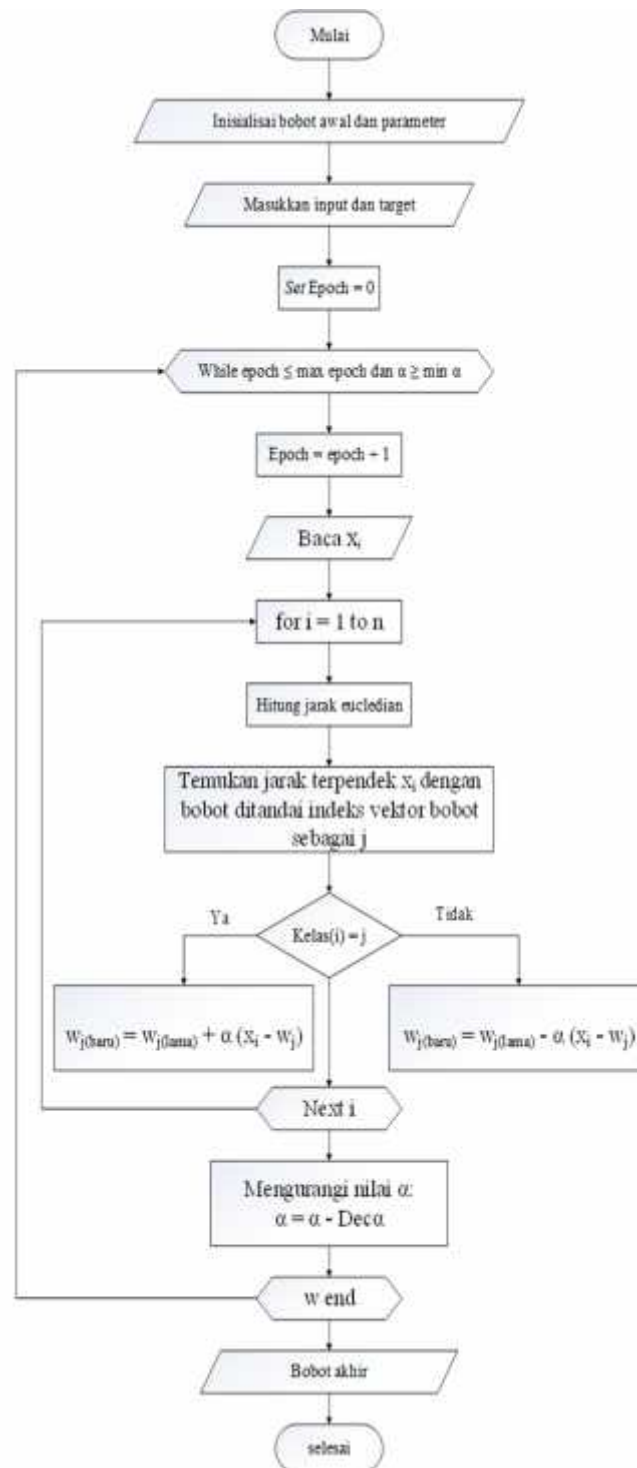


Gambar 2.9 Jaringan Learning Vector Quantization

Proses pembelajaran pada LVQ bertujuan menemukan nilai bobot yang mirip untuk mengklasifikasikan vektor – vektor input kedalam kelas yang sesuai dengan inisialisasi bobot awal pada saat pembentukan jaringan. Adapun parameter – parameter yang akan dipakai pada LVQ yaitu sebagai berikut (Hidayati dan Warsito 2010).

- a. α (*Learning Rate*), α . Jika α terlalu besar, algoritma akan menjadi tidak stabil. Sebaliknya, jika α terlalu kecil, prosesnya akan menjadi lama. Nilai α adalah $0 < \alpha < 1$.
- b. $D \propto$ (Penurunan *Learning Rate*), $D \propto$ adalah penurunan laju pembelajaran. Didefinisikan sebagai $(0.1 * \alpha)$.
- c. $M \propto$ (Minimum *Learning Rate*), $M \propto$ adalah nilai minimum laju pembelajaran yang masih dibolehkann.
- d. $M \quad h$ (Maksimum Epoch), $M \quad h$ adalah jumlah epoch atau iterasi maksimum yang dapat dilakukan selama pelatihan. Iterasi akan berhenti jika nilai epoch mencapai epoch maksimum.

Adapun algoritma proses pembelajaran pada metode LVQ dijelaskan pada gambar berikut ini.



Gambar 2.10 Algoritma proses pembelajaran pada metode LVQ

Jarak *euclidian* di lambangkan dengan D . Adapun rumus untuk menghitung D dapat dilihat pada persamaan berikut.

$$D = \sqrt{(x_1 - w_1)^2 + \dots + (x_n - w_n)^2}$$

Keterangan:

D adalah jarak *eucledian* antara data latih inputan dengan bobot kelas

x_1, x_n adalah nilai – nilai pada data latih inputan

w_1, w_n adalah bobot – bobot pada suatu kelas

Ketika jarak *eucledian* sudah didapatkan, maka akan dilakukan perbaikan terhadap bobot kelas yang memiliki jarak *eucledian* paling kecil. Dengan ketentuan sebagai berikut.

Jika kelas(i) = j, maka akan dilakukan perbaikan bobot seperti persamaan berikut.

$$W_{j(t+1)} = W_{j(t)} + \alpha (x_i - W_j)$$

Adapun jika kelas(i) \neq j, maka akan dilakukan perbaikan bobot seperti persamaan berikut.

$$W_{j(t+1)} = W_{j(t)} - \alpha (x_i - W_j)$$

Setelah dilakukan proses pembelajaran pada LVQ, akan didapatkan bobot akhir untuk setiap kelas. Bobot akhir inilah yang akan digunakan pada proses pengujian. Adapun alur proses pengujian pada LVQ dapat dilihat pada gambar berikut



Gambar 2.11 Algoritma proses pengujian pada metode LVQ

Pada tahap pengujian, data uji diuji dengan mencari jarak *euclidean* terdekat antara data uji dengan bobot akhir setiap kelas. Kelas dengan jarak yang terdekatlah yang menyatakan kelas data citra uji yang dimasukkan.

2.5.2 *Learning Vector Quantization 2.1 (LVQ 2.1)*

Modifikasi LVQ yang disebut LVQ2.1 (kohonen,1990a) mempertimbangkan dua vektor referensi terdekat, yaitu Y_{c1} dan Y_{c2} . Kondisi untuk memperbaharui kedua vektor tersebut yaitu jika salah satu dari vektor tersebut (misal, Y_{c1}) masuk ke dalam kelas yang sesuai dengan vektor masukan x , sementara vektor lainnya (misal, Y_{c2}) tidak masuk ke dalam kelas yang sesuai dengan vektor masukan x . Sebagaimana LVQ2, vektor x harus masuk ke dalam window supaya dapat terjadi pembaharuan. Window dinyatakan sebagai berikut :

$$\text{Min} [d_{(c1)}/d_{c2} , (d_{c2})/d_{c1}] > 1- \epsilon$$

$$\text{Max} [d_{(c1)}/d_{c2} , (d_{c2})/d_{c1}] < 1+ \epsilon$$

Jika kondisi-kondisi tersebut terpenuhi, maka vektor referensi yang masuk ke dalam kelas yang sesuai dengan vektor x akan diperbaharui dengan persamaan :

$$Y_{c1}(t+1) = Y_{c1}(t) + \alpha(t) [x(t) - Y_{c1}(t)]$$

dan vektor referensi yang tidak masuk ke dalam kelas yang sesuai dengan vektor x akan diperbaharui dengan persamaan :

$$Y_{c2}(t+1) = Y_{c2}(t) - \alpha(t) [x(t) - Y_{c2}(t)]$$

2.6 Pengenalan Tulisan Huruf t dengan Metode *Learning Vector Quantization 2.1*

Tahapan pengenalan tulisan tangan dengan metode *learning vector quantization 2.1* adalah sebagai berikut :

2.6.1 *Data Acquisition*

Data Acquisition merupakan tahap pengambilan citra menggunakan mesin scanner untuk melakukan scanning terhadap data tulisan tangan dari responden sehingga diperoleh citra digital tulisan.

2.6.2 *Labeling*

Setelah di peroleh data tulisan tangan dari responden , maka di lakukan proses *labeling* yang di lakukan oleh pakar grafologi. Proses ini bertujuan untuk menentukan karakter seseorang tersebut.

2.6.3 *Pre-processing*

Pada tahap pre-processing ini dilakukan proses perubahan warna pada citra yang dimasukkan dengan tahap pemisahan objek dan background dari citra tulisan tangan yang akan dikenali. Tahap pre-processing dilakukan untuk dapat mengubah ciri warna dari suatu citra sehingga dapat menjadi data input yang dapat dilanjutkan untuk tahap processing. Pada tahap pre-processing ini

dilakukan *cropping*, konversi RGB ke grayscale, konversi grayscale ke biner, dan resize.

2.6.3.1 Cropping

Cropping dilakukan adalah untuk meminimalisir background pada citra tulisan huruf t sehingga didapatkan area yang berfokus pada karakter huruf agar dapat diproses dengan baik pada tahap selanjutnya

2.6.3.2 Konversi RGB ke Grayscale

Salah satu persamaan dari konversi RGB ke grayscale yang digunakan adalah (Kadir dan Susanto, 2013):

$$\text{Grayscale} = (0,2989 * R) + (0,5870 * G) + (0,1141 * B) \dots\dots\dots (2.15)$$

Keterangan:

R = merepresentasikan nilai warna merah pada citra

G = merepresentasikan nilai warna hijau pada citra

B = merepresentasikan nilai warna biru pada citra

2.6.3.4 Resize

Pola tulisan yang telah digambarkan, yang telah dilakukan proses konversi seperti tahap sebelumnya maka akan dilakukan proses resize untuk mengubah ukuran semula menjadi matriks berukuran m x n piksel. Pada penelitian ini, aplikasi pengenalan karakter huruf t dengan menggunakan matriks 20 x 20 piksel.

2.7 Processing

Setelah tahapan pre-processing selesai dilakukan, maka tahapan selanjutnya adalah processing yang berfungsi untuk mengekstraksi fitur (feature extraction) agar diperoleh sebuah vektor dari suatu image. Ekstraksi ciri citra

dikerjakan dengan mengubah dimensi suatu citra menjadi satu kolom sehingga menjadi vektor matriks untuk suatu citra. Vektor-vektor ini kemudian dikelompokkan menjadi satu matriks besar sebagai masukan untuk proses ekstraksi ciri. Proses berikutnya adalah menemukan matriks rata-rata dari data latih. Matriks ini kemudian dihitung rata-rata tiap barisnya, kemudian dikurangkan tiap elemen matriks dengan rata-rata tiap baris.

2.8 Classification

Setelah tahapan processing selesai dilakukan, maka tahapan selanjutnya adalah classification. Tahapan-tahapan yang telah dilakukan sebelumnya bertujuan untuk menyederhanakan agar dapat diproses pada metode *Learning Vector Quantization*.

2.9 Pengujian

Pengujian yang dilakukan adalah pengujian black box untuk melihat apakah aplikasi yang dibuat telah sesuai dengan analisa dan perancangan yang telah dilakukan sebelumnya.

2.9.1 Pengujian Black Box

Pengujian black box atau behavioral adalah pemeriksaan mengenai sesuatu yang dilakukan aplikasi, terutama perilakunya dan masalah-masalah bisnis. Black Box digunakan untuk mengidentifikasi kesalahan (*bug*) yang terdapat pada hasil-hasil, pemrosesan dan perilaku dari sistem. (Black, 2007). Pengujian black box mengkhususkan pada perilaku keluaran dari suatu aplikasi komponennya sambil melihat objek yang diuji sebagai suatu kotak hitam (black box) sehingga mencegah tester untuk melihat isi di dalamnya. Pengujian black box memverifikasi penanganan yang benar dari fungsi-fungsi keluaran yang disediakan oleh aplikasi atau apakah memenuhi harapan.

2.9.2 Akurasi dan Tingkat Error

Tingkat keberhasilan sistem dapat dihitung berdasarkan perbandingan jumlah klasifikasi yang berhasil terhadap seluruh jumlah data yang diuji. Sehingga menghasilkan persamaan:

$$\text{Akurasi} = \frac{\sum P}{\sum B} \times 100\%$$

$$E = 100\% - a$$

2.9.3 Pengujian Karakter

Pengujian karakter ini dilakukan dengan membandingkan tulisan huruf t yang sudah dianalisa oleh sistem dengan tulisan huruf t yg sudah di label oleh pakar grafologi.

2.10 Penelitian Terkait

PENELITI	JUDUL	METODE	KESIMPULAN
Miftah Ari (2012)	Perbandingan Antara Metode Backpropagation Dan Learning vector quantization (LVQ) Pada Pengenalan Citra Barcode	Backpropagation Dan Learning vector quantization	nilai akurasi metode learning vector quantization menghasilkan 94% lebih baik dibandingkan dengan metode backpropagation yang hanya menghasilkan 75,5%

Suri (2016)	Diagnosa Penyakit <i>Cerebrovascular Disease</i> (CVD) atau Stroke Menggunakan Metode <i>Learning Vector Quantization 2.1</i>	<i>Learning Vector Quantization 2.1</i>	Pada penelitian ini berdasarkan Nilai ketidakpastian Sistem Pakar dengan persentase akurasi yang sempurna yaitu 100%
Elvia Budianita dan Muhammad Firdaus (2016)	diagnosis penyakit kejiwaan menggunakan LVQ2	Learning Vector Quantization 2	hasil dari penelitian yang telah diujikan diperoleh hasil nilai akurasi sebesar 90% yakni hasil perbandingan antara jumlah data yang benar dengan jumlah data uji