

BAB II TINJAUAN PUSTAKA

Bab ini membahas kajian pustaka dan dasar teori yang digunakan untuk menunjang penulisan skripsi mengenai Implementasi Metode *Support Vector Machine* dengan *Fuzzy Analytic Hierarchy Process* pada Sistem Pendukung Keputusan Untuk Seleksi Penerimaan *Student Employee* (Studi Kasus Program Teknologi Informasi Dan Ilmu Komputer). Beberapa dasar teori yang dimaksud adalah Sistem Pendukung Keputusan (SPK), *Student Employee*, *Support Vector Machine* (SVM), algoritma *Sequential Training SVM*, *Analytic Hierarchy Process* (AHP), dan *Fuzzy Analytic Hierarchy Process* (F-AHP).

2.1. Kajian Pustaka

Klasifikasi merupakan salah satu bentuk peramalan yang memiliki nilai keluaran diskrit, dan bertujuan untuk menemukan suatu fungsi keputusan $f(x)$ yang secara akurat memprediksi kelas dari data. Penelitian oleh Widyarini mengatakan bahwa *Support vector machines* (SVM) merupakan suatu metode yang handal dalam menyelesaikan masalah klasifikasi data. Permasalahan SVM dipecahkan dengan menyelesaikan persamaan Lagrangian yang merupakan bentuk dual dari SVM melalui *quadratic programming*. Melalui pendekatan SVM standar ini, waktu komputasi yang cukup panjang diperlukan untuk mendapatkan solusi optimal [SAN-09].

Berikut kajian pustaka yang menjelaskan beberapa penelitian yang telah dilakukan terkait dengan klasifikasi pada sistem pendukung keputusan yang ditunjukkan melalui Tabel 2.1. Diantaranya yaitu penelitian yang dilakukan oleh Widyarini dalam penggunaan metode *Support Vector Machine*.

Widyarini mengaplikasikan metode *Cross Entropy* untuk *Support Vector Machine*. *Cross Entropy* (CE) merupakan suatu metode optimasi yang baru dikembangkan dengan dua prosedur utama, yaitu melakukan *generate* data sampel dengan distribusi tertentu dan melakukan *update* parameter distribusi berdasarkan data sampel elite untuk menghasilkan sampel yang lebih baik pada iterasi berikutnya [SAN-09].

Tabel 2.1. Kajian Pustaka

No	Judul	Objek	Metode	Hasil
1	Aplikasi Metode <i>Cross Entropy</i> untuk <i>Support Vector Machines</i> [SAN-09]	<p>Objek: Menyelesaikan masalah optimasi <i>lagrange</i> SVM agar didapatkan proses komputasi yang lebih cepat dan sederhana menggunakan metode <i>Cross Entropy</i></p> <p>Input: 6 kumpulan <i>dataset</i> dari kasus nyata yaitu data <i>Hepatitis, Iris, Breast Cancer, Haberman's Survival, Credit Approval, dan Splice</i></p> <p>Parameter:</p> <ul style="list-style-type: none"> - Bilangan <i>Random</i> (N)=40 - P = 0,2 (20% data sampel) - $\alpha(\sigma) = 10^{-8}$ - C = 2 - $\sigma = 2$ 	<p>Langkah dan metode:</p> <ol style="list-style-type: none"> 1. Menghitung matriks Kernel 2. Menetapkan parameter awal yaitu: N, C, ρ 3. Men-<i>generate</i> sampel α 4. Menghitung solusi Lagrangian SVM 5. Mengurutkan Lagrangian SVM 6. Mengambil sampel <i>elite</i> 7. Melakukan <i>update</i> parameter distribusi 8. Mengulang langkah 3 hingga 7 9. Menetapkan nilai α 	Metode SVM-CE yang dikembangkan terbukti mampu mengklasifikasi data dengan akurasi yang sebanding dengan SVM standar dan Kernel Adatron (KA). Dari output yang dihasilkan dapat dilihat bahwa semakin besar jumlah data, kinerja SVM-CE akan semakin baik dari segi kecepatan waktu komputasi dibandingkan dengan KA dan SVM standar.
2	Penerapan Metode AHP pada Transaksi Jual-Beli di Jejaring Komunitas Pecinta Reptil [REP-12]	<p>Objek: Merancang sistem untuk melakukan proses pendukung keputusan pemilihan hewan menggunakan</p>	<p>Langkah dan metode:</p> <ol style="list-style-type: none"> 1. Menentukan jenis ular dan kriteria-kriteria 	Jenis ular Boa adalah jenis ular yang memiliki nilai paling tinggi diantara alternatif lainnya dan Boa sesuai dengan kriteria-

		<p><i>Analytical Hierarchy Process (AHP)</i> Input: Nilai prioritas masing-masing kriteria yaitu,</p> <ul style="list-style-type: none"> - Harga - Kesehatan - Corak - Ukuran - Karakter <p>Parameter:</p> <ul style="list-style-type: none"> - <i>Lambda</i> Maksimum - CI - CR 	<ol style="list-style-type: none"> 2. Menentukan jenis kriteria dalam bentuk matriks berpasangan 3. Menentukan prioritas kriteria 4. Menentukan jenis ular dalam bentuk matriks berpasangan 5. Menentukan prioritas jenis ular 6. Mengukur konsistensi 7. Menghitung prioritas global 	<p>kriteria yang diberikan sebelumnya. Uji konsistensi menunjukkan nilai Rasio Inkonsistensi kurang dari 0,1(10%)</p>
3.	<p>Sistem Pendukung Keputusan Pemilihan Karyawan Terbaik Menggunakan Metode <i>Fuzzy Analytic Hierarchy Process (F-AHP)</i> [JAS-11]</p>	<p>Objek: Membangun SPK pemilihan karyawan terbaik menggunakan metode <i>Fuzzy AHP (F-AHP)</i> Input: Nilai prioritas kriteria yaitu,</p> <ul style="list-style-type: none"> - Penilaian lingkungan kerja: Kooperatif, Kualitas Kerja, dan Manajerial - Penilaian Konsumen: Kenyamanan, Kualitas Layanan, dan Informatif 	<p>Langkah dan metode:</p> <ol style="list-style-type: none"> 1. Membuat struktur hirarki masalah dan menentukan perbandingan matriks berpasangan antar kriteria dengan skala <i>Triangular Fuzzy Number (TFN)</i> 2. Menentukan nilai sintesis fuzzy (Si) prioritas 3. Menentukan nilai vektor (V) dan nilai ordinat defuzzifikasi (d') 	<p>Menghasilkan keputusan yang lebih objektif berupa tabel perankingan terhadap 5 orang karyawan berdasarkan bobot global.</p>

		<ul style="list-style-type: none"> - Sikap dan kepribadian: Disiplin, Spirit/Motivasi, dan Tanggung Jawab - SOP/Teknis: Kecepatan Layanan, Ketelitian, dan Pemahaman/Keahlian 	<p>4. Normalisasi nilai bobot vektor <i>fuzzy</i> (W)</p>	
4	<p><i>An Integrated SVM and Fuzzy AHP Approach for Selecting Third Party Logistics Providers</i> [TPL-12]</p>	<p>Objek: Mengintegrasikan metode SVM dan <i>Fuzzy AHP</i> untuk seleksi penyedia logistik pihak ketiga.</p> <p>Input:</p> <ul style="list-style-type: none"> - <i>Service level</i> - <i>Technical level</i> - <i>Cost level</i> - <i>Operating efficiency</i> - <i>Financial Situation</i> respectively 	<p>Langkah dan metode:</p> <ol style="list-style-type: none"> 1. Klasifikasi nonlinear 2. Fungsi <i>Decision-making</i> 3. Mencari <i>hyperplane</i> yang paling optimal 4. Menggunakan metode <i>Lagrange</i> 5. Fungsi <i>Decision-making</i> dengan kernel $K(x_i, x_j)$, α adalah <i>Lagrange multiplier</i> 6. Terbentuk solusi optimal; alpha dan nilai bias 7. Kalkulasi pembobotan menggunakan <i>triangular fuzzy number</i>. 8. Mencari nilai W_i 	<p>SVM-FAHP dapat meningkatkan efisiensi seleksi dan mengurangi biaya komputasi selama proses pengambilan keputusan dan biaya pengumpulan informasi. SVM menjamin keakuratan klasifikasi dan model FAHP dapat memecahkan masalah ketidakpastian efektif ketika mengkonversi kasus kualitatif yang kuantitatif</p>

Sumber : [SAN-09][REP-12][JAS-11][TPL-12]

Pada penelitian Widyarini metode CE diterapkan untuk menyelesaikan masalah optimasi *lagrange* SVM agar didapatkan proses komputasi yang lebih cepat dan sederhana. Nilai α pada *lagrange* SVM akan dicari dan di-update menggunakan metode CE untuk menghasilkan *clasifier* yang paling optimal [SAN-09]. Metode CE menyelesaikan suatu permasalahan optimasi dengan mula-mula membangkitkan bilangan random dalam distribusi tertentu, kemudian dari sampel tersebut dipilih sampel terbaik untuk menjadi parameter bilangan random pada iterasi berikutnya. Pada penelitian ini, sampel yang dibangkitkan adalah nilai α pada *lagrange* SVM. Algoritma yang diusulkan dijelaskan pada langkah satu hingga Sembilan [SAN-09].

1. Menghitung matriks Kernel dikalikan label keluaran (*output*) dari data *training*.
2. Menetapkan parameter awal
Parameter yang diperlukan yaitu jumlah sampel random yang akan dibangkitkan (N), batasan nilai α yang diijinkan dalam pembuatan sampel (C), persentase sampel *elite* (ρ).
3. Men-generate sampel α
Nilai *Lagrange multiplier* α_i , untuk $i = 1$ hingga m di-generate menggunakan distribusi tertentu sebanyak N, dengan batasan nilai $0 \leq \alpha \leq C$. Pada penelitian ini digunakan distribusi normal.
4. Menghitung solusi Lagrangian SVM
Untuk semua nilai α dari $l = 1$ hingga N, hitung solusi Lagrangian SVM $W(\alpha)$.
5. Mengurutkan Lagrangian SVM
6. Mengambil sampel *elite*
Sampel *elite* diambil dari $W(\alpha)$ sebanyak ρN dengan nilai $W(\alpha)$ terbesar setelah diurutkan.
7. Melakukan *update* parameter distribusi
Pada penelitian ini, distribusi yang digunakan adalah distribusi normal dengan parameter rata-rata (μ) dan standar deviasi (σ^2). Jika i adalah indeks dari N^{elite} sampel *elite* dan θ adalah variabel *smoothing*. Digunakan $\theta = 1$.
8. Mengulang langkah 3 hingga 7

Pengulangan dilakukan hingga *stopping criterion* terpenuhi. *Stopping criterion* yang digunakan dalam penelitian ini adalah ketika standar deviasi maksimum dari α telah kurang dari 10^{-8} .

9. Menetapkan nilai α . Untuk distribusi normal, nilai α adalah rata-rata α (μ) yang diperoleh dari iterasi terakhir.

Untuk mendapatkan fungsi *clasifier*, hitung vektor w optimal dengan menggunakan α yang didapatkan dari langkah 9 dan hitung bias. Cari label data testing menggunakan persamaan $F(x) = \text{sign}((w^*,x)+b)$, yakni pembulatan hasil dari perkalian vektor w^* dengan data testing x ditambah bias [SAN-09].

Data yang digunakan untuk pengujian terdiri atas 6 kumpulan *dataset* dari kasus nyata yaitu data Hepatitis, Iris, Breast Cancer, Haberman's Survival, Credit Approval, dan Splice. Parameter komputasi CE yang digunakan adalah dengan melakukan *generate* 40 bilangan random ($N=40$) dan penggunaan nilai ρ sebesar 0,2 (20% sampel). Tiap metode diuji dengan persentase data training dan data testing sebesar 2:1, dengan 20 kali pengujian. Untuk iterasi pada SVM-CE, *stopping criterion* yang digunakan adalah ketika nilai maksimum dari standar deviasi $\alpha(\sigma)$ kurang dari 10^{-8} . Biaya SVM yang dikenakan adalah $C = 2$. Metode Kernel yang digunakan adalah Kernel RBF dengan $\sigma = 2$, Kernel Polynomial Derajat 5, serta Kernel linear. Dilihat dari hasil pengujian pada keenam dataset, metode SVM-CE yang dikembangkan terbukti mampu mengklasifikasi data dengan akurasi yang sebanding dengan SVM standar dan Kernel Adatron (KA) [SAN-09]. Metode ini dapat digunakan untuk penelitian dengan data berukuran besar, dimana dibutuhkan kecepatan komputasi dengan akurasi yang tetap terjaga, karena metode ini telah terbukti mempunyai waktu komputasi yang cepat terutama untuk permasalahan dengan data yang besar. Sehingga metode CE tidak digunakan pada penelitian ini karena akurasi yang dihasilkan pada penelitian Widyarini sebanding dengan metode SVM standar, tidak lebih baik dari SVM standar dan metode SVM-CE cocok digunakan pada data yang berukuran besar sedangkan pada penelitian ini tidak menggunakan data yang berukuran besar.

Pada penelitian Prima, metode *Analytic Hierarchy Process* (AHP) digunakan pada sistem pemilihan reptil dalam suatu jejaring sosial. Fasilitas dalam jejaring interaktif yang diperuntukkan bagi pecinta reptile ini yaitu memilih atau

memberikan rekomendasi reptile mana yang tepat berdasarkan prioritas kriteria tertentu. Dalam penelitian tersebut menggunakan 5 kriteria sebagai bahan pertimbangan untuk memilih reptile yang sesuai. Terdapat 3 jenis reptil yang akan dipilih, colubrid, phyton, dan boa. Pada langkah awal, Prima menentukan jenis kriteria dan menyusunnya dalam bentuk matriks berpasangan pada level 1. Adapun langkah-langkah penelitian menggunakan AHP sebagai berikut:

1. Menentukan jenis Ular pada forum di jejaring interaktif ini. Kriteria-kriteria yang diperlukan (dalam proyek akhir ini ada 5 kriteria) dan nama jenis ular tersebut adalah colubrid, phyton, boa.
2. Menentukan jenis-jenis kriteria yang akan menjadi persyaratan pemilihan jenis ular dan menyusun kriteria-kriteria tersebut dalam bentuk matriks berpasangan

Cara pengisian elemen-elemen matriks pada tabel:

- a. Elemen $a[i,j] = l$ dimana $l = 1,2,\dots,n$. (untuk penelitian ini $n = 7$).
 - b. Elemen Matriks Segitiga atas sebagai input.
 - c. Elemen Matriks segitiga bawah mempunyai rumus $a[j,i] = 1/a[i,j]$ untuk $i \neq j$.
3. Menjumlahkan setiap kolom pada tabel
 4. Menentukan nilai elemen kolom kriteria dengan rumus : tiap-tiap sel pada tabel dibagi dengan masing-masing jumlah kolom pada langkah 3.
 5. Menentukan Prioritas Kriteria pada masing-masing baris pada tabel dengan rumus jumlah baris dibagi banyak kriteria.
 6. Memasukkan nama-nama jenis ular dalam bentuk matriks berpasangan.
 7. Menjumlahkan setiap kolom pada tabel.
 8. Menentukan nilai elemen kolom nama dengan rumor : tiap-tiap sel pada tabel dibagi dengan masing-masing jumlah kolom pada langkah 7.
 9. Menentukan Prioritas Calon pada masing-masing baris pada tabel dengan rumus jumlah baris dibagi banyak calon.
 10. Mengukur konsistensi, yaitu menguji konsistensi.
 11. Menghitung λ Max, CI, dan CR
 12. Menghitung nilai prioritas global.

Prima membuat matriks berpasangan dengan nama setiap alternatif (jenis reptil). Ada 3 jenis ular yang digunakan. Masing-masing kolom pada tabel matriks level 2 diberikan nilai perbandingan berdasarkan nilai dari setiap kriteria yang ada pada level 1. Lima tabel matriks pada level 2 menghasilkan nilai bobot dari masing-masing alternatif (jenis reptil) [REP-2011].

Didapat bahwa BOA adalah jenis ular yang memiliki nilai paling tinggi diantara alternative lainnya, dengan demikian BOA sesuai dengan kriteria-kriteria yang diberikan sebelumnya. Kemudian memeriksa nilai konsistensi dengan mencari nilai Lambda Max, CI, dan CR [REP-2011]. Penelitian tersebut menunjukkan metode AHP memiliki CR (*Consistency Ratio*) baik yaitu kurang dari 0,1 [REP-2011].

Fuzzy AHP merupakan pengembangan metode AHP dengan menggunakan pendekatan konsep *fuzzy*. *Fuzzy Analytic Hierarchy Process* (F-AHP) menutupi kelemahan yang terdapat pada AHP, yaitu permasalahan terhadap kriteria yang memiliki sifat subjektif lebih banyak. Dalam penelitian Jasril metode F-AHP digunakan untuk membangun sistem pendukung keputusan pemilihan karyawan terbaik. Adapun langkah-langkah penyelesaian F-AHP yang digunakan oleh Jasril sebagai berikut [JAS-11]:

- a. Membuat struktur hirarki masalah yang akan diselesaikan dan menentukan perbandingan matriks berpasangan antar kriteria dengan skala TFN.
- b. Menentukan nilai sintesis *fuzzy* (S_i) prioritas
- c. Menentukan nilai vektor (V) dan nilai ordinat defuzzifikasi (d').
- d. Normalisasi nilai bobot vektor *fuzzy* (W)

Setelah dilakukan normalisasi dari persamaan W' maka nilai bobot vektor yang ternormalisasi adalah $W (d(A_1), d(A_2), \dots, d(A_n))^T$ dimana W adalah bilangan non *fuzzy*.

Alasan digunakannya metode F-AHP pada penelitian ini karena metode F-AHP dinilai mampu menyelesaikan kasus *multi-decision* yang nantinya akan diterapkan pada sistem pendukung keputusan yang bercabang.

Support Vector Machine (SVM) menjamin akurasi dari klasifikasi dan FAHP menyelesaikan masalah secara efektif. Penelitian oleh Guiyun Liu

menunjukkan bahwa model SVM-FAHP dapat dikerjakan dengan mudah dan lebih efektif pada pemilihan penyedia logistik pihak ketiga (*Third Party Logistics Providers/3PL*). Pada proses awal dilakukan klasifikasi data menggunakan metode SVM dengan algoritma *one-to-one multiclassification*. Data diklasifikasikan menjadi 4 kelas yaitu, *poor*, *medium*, *good*, dan *excellent*. Data yang digunakan sebanyak 50 data yaitu 40 data *training* dan 10 data *testing*. Langkah pertama yaitu melakukan klasifikasi data (40) ke dalam kelas *excellent* (*top 10*) dan *non-excellent* (30). Pada kelas *excellent* ditandai dengan 1 dan *non-excellent* dengan -1. Kedua melakukan klasifikasi data *non-excellent* (30) ke dalam kelas *good* (*top 10*) dan *non-good* (20). Pada kelas *good* ditandai dengan 1 dan *non-good* dengan -1. Ketiga melakukan klasifikasi data *non-good* (30) ke dalam kelas *medium* (*top 10*) dan *non-medium* (20). Pada kelas *medium* ditandai dengan 1 dan *non-medium* dengan -1. Model multi-classification ditentukan berdasarkan proses tersebut dan melakukan penghitungan pada parameter *nsv1*, *nsv2*, *nsv3*, *alpha1*, *alpha2*, *alpha3*, *bias1*, *bias2*, *bias3* [TPL-2012].

Berdasarkan hasil survey dan membuat setiap kriteria berdasarkan penentuan sistem indeks sebelumnya, maka Guiyun Liu memperoleh 30 data *sample*. Pertama memasukkan 30 grup data ke dalam fungsi yang memiliki parameter *nsv1*, *alpha1*, *bias1*, menghasilkan data keluaran sebanyak 7 data yang masuk ke dalam kelas 1 (*excellent*), dan yang lain ke dalam kelas -1 (*non-excellent*). Data dari kelas -1 (*non-excellent*), 23 data, dimasukkan dalam fungsi yang memiliki parameter *nsv2*, *alpha2*, *bias2*, menghasilkan sebanyak 8 data yang masuk ke dalam kelas 1 (*good*), dan yang lain ke dalam kelas -1 (*non-good*). Data dari kelas -1 (*non-good*), 15 data, dimasukkan dalam fungsi yang memiliki parameter *nsv3*, *alpha3*, *bias3*, menghasilkan sebanyak 8 data yang masuk ke dalam kelas 1 (*medium*), dan yang lain ke dalam kelas -1 (*non-medium*).

Dimisalkan training sample set adalah sebagai berikut $\{(x_i, y_i), i = 1, 2, \dots, l\}$. Sample ini berisi 2 jenis data, ketika sample i adalah kelas pertama maka $y_i = 1$, sedangkan jika sample i adalah kelas kedua maka $y_i = -1$. Berikut fungsi *decision-making* yang memisahkan 2 jenis data [TPL-2012]:

1. Menetapkan klasifikasi adalah klasifikasi nonlinear
2. Menetapkan awal fungsi *decision-making*:

3. Penyelesaian masalah *hyper-plan* yang optimal
4. Menggunakan metode *Lagrange* untuk penjabaran fungsi
5. Menetapkan fungsi *decision-making* setelah kernel terbentuk.

$K(x_i, x_j)$ merupakan fungsi kernel, α adalah *Lagrange multiplier*, dan C adalah parameter control.

6. Menetapkan solusi optimal

Dari studi Hilbert-Schmidt, fungsi simetris dapat sebagai fungsi kernel.

Secara umum fungsi kernel adalah sebagai berikut:

- *Polynomial Kernel Function*
- *Radial Basis Function (RBF)*
- *Sigmoid Kernel Function*

Tujuh data yang masuk ke kelas *excellent* pada hasil keluaran metode SVM dijadikan sebagai data masukan pada tahap pembobotan menggunakan metode FAHP [TPL-2012]. Aspek yang dianalisa untuk menghitung bobot indeks menggunakan metode penilaian matriks *fuzzy* ada 5 yaitu *customer service level*, *technical level*, *cost level*, *operation efficiency*, dan *financial situation*. Langkah-langkah yang digunakan untuk metode F-AHP adalah sebagai berikut:

1. Langkah pertama menentukan perbandingan matriks berpasangan antar kriteria skala TFN (*Triangular Fuzzy Number*) pada matriks M .
2. Membuat bobot kriteria dengan membuat matriks W . Matriks M di definisikan kedalam 3 matriks yaitu l , m , dan u .
3. Melakukan normalisasi pada matriks W_i ($i = 1,2,3,4,5$)

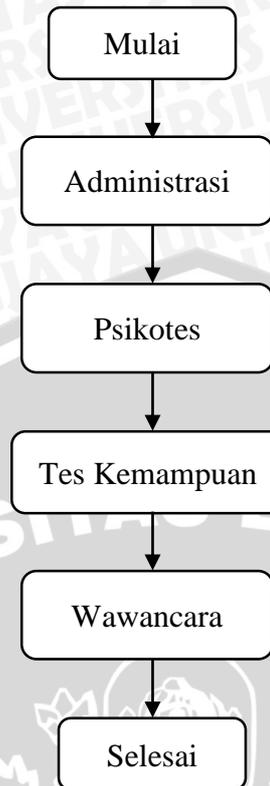
Dalam penelitian Liu bobot normalisasi matriks W_i ($i=1,2,3,4,5$) dan didapatkan hasil bobot yaitu $W_1 = 0,0526$, $W_2 = 0,3495$, $W_3 = 0,1933$, $W_4 = 0,3029$, $W_5 = 0,1017$. Perbandingan matriks untuk *customer service level* (A), *technical level* (B), *cost level* (C), *operation efficiency* (D), dan *financial situation* (E) dapat ditentukan secara berurutan. Berdasarkan 7 data *sample* yang masuk dalam kelas *excellent*, skor evaluasi dikalkulasi menggunakan metode pembobotan sehingga hasil akhir berupa bobot index dari 7 data tersebut yang diurutkan berdasarkan bobotnya. $S_1 = 66,1207$, $S_2 = 67,2280$, $S_3 = 69,6905$, $S_4 = 69,7883$, $S_5 = 66,8707$, $S_6 = 70,3667$, $S_7 = 69,0050$. *Third party logistic (3PL) provider* yang memiliki bobot tertinggi adalah 3PL *provider* terbaik [TPL-2012]. Model yang didasarkan

pada SVM-FAHP dapat meningkatkan efisiensi seleksi dan mengurangi biaya komputasi selama proses pengambilan keputusan dan biaya pengumpulan informasi. SVM menjamin keakuratan klasifikasi karena SVM dapat melakukan *learning* dari sampel kecil secara akurat. Model F-AHP dapat memecahkan masalah ketidakpastian efektif ketika mengkonversi kasus kualitatif yang kuantitatif. Contoh penelitian menunjukkan bahwa model SVM-FAHP layak dan efektif [TPL-2012].

2.2. Student Employee

Student Employee merupakan seorang mahasiswa yang bekerja secara paruh waktu dan dipekerjakan sesuai dengan program (*Student Employee*) yang dilakukan oleh Universitas tertentu [MCG-14]. Universitas Brawijaya (UB) sendiri yaitu pada Program Teknologi Informasi dan Ilmu Komputer (PTIHK) telah mencanangkan dan melaksanakan program *Student Employee* untuk mahasiswa PTIHK. Menurut Kasyful Amron, ST., M.Sc selaku pengurus program *Student Employee* PTIHK UB dan sebagai kepala BPTIK mengatakan bahwa tujuan dilaksanakannya program dengan masa jabatan 1 tahun bekerja paruh waktu di PTIHK UB ini agar mahasiswa dapat ikut serta membantu menyelesaikan kegiatan atau penelitian terkait dengan pengembangan teknologi yang dilakukan oleh Badan Pengembangan Teknologi dan Ilmu Komputer (BPTIK) PTIHK UB. Dalam program ini tentunya akan dipilih mahasiswa yang dinilai cukup berkompeten pada bidang-bidang yang ditawarkan. BPTIK UB memiliki 4 bidang kerja *Student Employee* sebagai pilihan yaitu, *Programmer*, *Web Design*, *Multimedia*, dan *Network Admin*. Setiap bidang memiliki standar kualifikasi sesuai dengan parameter yang digunakan masing-masing. Standar kualifikasi ini digunakan untuk menyeleksi dan memilih mahasiswa yang dinilai cukup berkompeten serta mampu bekerja sesuai harapan BPTIK UB.

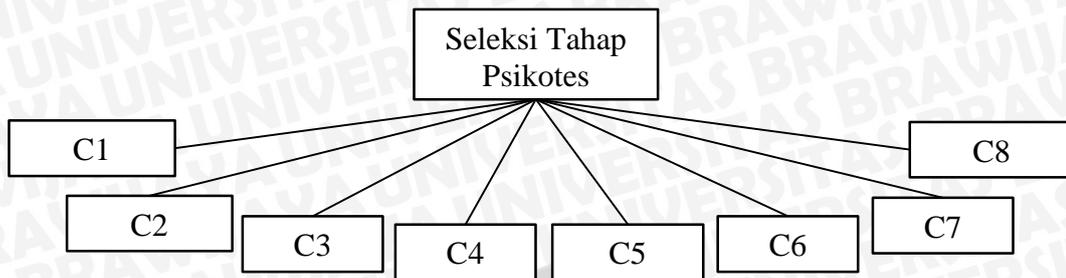
Proses seleksi yang dimiliki BPTIK UB untuk program *Student Employee* memiliki 4 tahapan yaitu Administrasi, Psikotes, Tes Kemampuan, dan Wawancara. Tahapan-tahapan tersebut dapat ditunjukkan pada Gambar 2.1. Masing-masing tahapan diberlakukan sistem gugur yang artinya apabila mahasiswa tidak lolos pada tahap tertentu maka dia tidak bisa melanjutkan ke tahap seleksi berikutnya.



Gambar 2.1. Tahapan Proses Seleksi *Student Employee*
Sumber: [Wawancara]

Pada tahap pertama adalah tahap administrasi yaitu mahasiswa diminta untuk mengumpulkan data diri, nilai akademik, serta menerangkan kemampuan dan bidang yang diminati. Data mahasiswa yang masuk kemudian diseleksi dan mahasiswa yang lolos akan lanjut ke tahap psikotes atau tes psikologi. Pada tahap psikotes ini mahasiswa akan menjalani tes intelegensi dan kepribadian. Tahap ketiga adalah tes kemampuan yaitu mahasiswa diminta menyelesaikan beberapa kasus sesuai dengan bidang yang diminati. Hasil pengerjaan pada tes kemampuan dinilai berdasarkan parameter yang ada pada tiap-tiap bidang. Tahap yang terakhir adalah seleksi wawancara yaitu mahasiswa yang lolos tes kemampuan akan menjalani sesi wawancara dengan tim penilai. Mahasiswa yang dapat melewati tiap tahapan hingga selesai akan diterima sebagai *Student Employee* BPTIK UB untuk masa jabatan 1 tahun kedepan.

Berikut penggambaran kriteria penilaian pada tahap psikotes dapat ditunjukkan pada Gambar 2.2.



Gambar 2.2. Kriteria Penilaian Tahap Psikotes
Sumber: [Wawancara]

Penjelasan untuk kriteria yang akan digunakan pada tahap psikotes tersebut (pada tahap psikotes) dapat dilihat pada Tabel 2.2.

Tabel 2.2. Kriteria Pengambilan Keputusan Seleksi Tahap Psikotes

Kriteria		Penjelasan
C1	Introvert	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Introvert</i>
C2	Ekstrovert	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Ekstrovert</i>
C3	Sensing	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Sensing</i>
C4	Intuiton	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Intuition</i>
C5	Thinking	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Thinking</i>
C6	Feeling	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Feeling</i>
C7	Judging	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Judging</i>
C8	Perceiving	Nilai dari hasil psikotes peserta seleksi yang masuk pada kategori <i>Perceiving</i>

Sumber: [Wawancara]

Berikut ini diuraikan kondisi masing-masing kriteria, dan rentang nilai yang digunakan dalam masing-masing kriteria [Wawancara]:

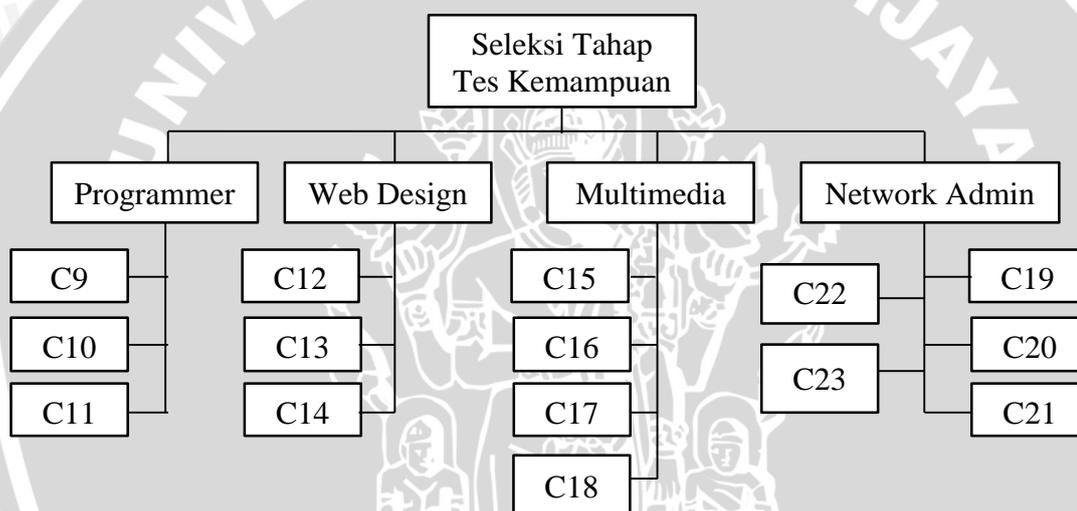
- Nilai Introvert

Nilai *introvert* ini didapatkan dari hasil psikotes peserta seleksi. Introvert merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih suka memikirkan dirinya dari pada orang lain.

- Nilai Ekstrovert
Nilai *ekstrovert* ini didapatkan dari hasil psikotes peserta seleksi. Ekstrovert merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang berinteraksi dengan orang lain.
- Nilai Sensing
Nilai *sensing* ini didapatkan dari hasil psikotes peserta seleksi. Sensing merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih suka menggunakan kekuatan otot dari pada otak.
- Nilai Intuition
Nilai *intuition* ini didapatkan dari hasil psikotes peserta seleksi. Intuition merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih berimajinasi atau berfantasi.
- Nilai Thinking
Nilai *thinking* ini didapatkan dari hasil psikotes peserta seleksi. Thinking merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih suka berfikir.
- Nilai Feeling
Nilai *feeling* ini didapatkan dari hasil psikotes peserta seleksi. Feeling merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih suka menggunakan perasaan, selalu memikirkan perasaan orang lain, dan mudah iba bila melihat orang lain kesusahan.
- Nilai Judging
Nilai *judging* ini didapatkan dari hasil psikotes peserta seleksi. Judging merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih tergantung pada penilaian atau penampilan luar, sistematis, berfikir, dan berperilaku secara teratur.
- Nilai Perceiving
Nilai *ekstrovert* ini didapatkan dari hasil psikotes peserta seleksi. Ekstrovert merupakan parameter dengan rentang nilai 0-100 yang memprediksi seberapa tinggi kadar seseorang lebih suka bersikap fleksibel, spontan,

adaptif, dan bertindak secara acak untuk melihat beragam peluang yang muncul. Tabel 4.4 menunjukkan 33 data yang akan diolah pada proses perhitungan.

Proses setelah tahap psikotes yaitu tahap tes kemampuan dihitung menggunakan metode FAHP untuk mendapatkan skor masing-masing alternatif. Proses ini dilakukan pada tahap tes kemampuan. Data yang digunakan pada proses ini adalah data yang lolos seleksi pada tahap psikotes. Pada tahap ini proses perhitungan dilakukan di masing-masing bidang yang berbeda yaitu: programmer, web design, multimedia, dan network admin dan setiap bidang memiliki kriteria yang berbeda-beda. Kriteria masing-masing bidang dapat ditunjukkan pada Gambar 2.3.



Gambar 2.3. Kriteria Penilaian Setiap Bidang Penempatan
Sumber:[Wawancara]

Penjelasan untuk kriteria yang akan digunakan pada tahap tes kemampuan (menggunakan metode FAHP) dapat dilihat pada Tabel 2.3 berikut.

Tabel 2.3. Kriteria Pengambilan Keputusan pada Tahap Tes Kemampuan

Kriteria		Penjelasan
C9	Mahir PHP & HTML	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Programmer</i>
C10	Menguasai Javascript & CSS	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Programmer</i>
C11	Kemampuan MySQL & SQL Server	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Programmer</i>

C12	Menguasai HTML & Javascript	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Web Design</i>
C13	Berpengalaman Blender3D/PHP MySQL	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Web Design</i>
C14	Menguasai Adobe Photoshop, Dreamweaver, Illustrator	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Web Design</i>
C15	Menguasai <i>Network Component, Network OS, Basic Computer, Hardware Component</i>	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Network Admin</i>
C16	Memahami TCP/IP, LAN, WAN, Switches, Router <i>Configuration</i>	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Network Admin</i>
C17	Familiar dengan Windows dan Linux	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Network Admin</i>
C18	Familiar dengan IP Network & Fiber Network	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Network Admin</i>
C19	Pengalaman menggunakan <i>Popular Network Product</i>	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Network Admin</i>
C20	Menguasai & memahami dasar Fotografi & Videografi	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Multimedia</i>
C21	Memahami teknik pembuatan skenario	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Multimedia</i>
C22	Menguasai perangkat Audio Video Editing: Adobe Premiere, After Affect, DJ Studio, Sony	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Multimedia</i>

C23	Mampu berfikir kreatif & Mencipta ide-ide baru	Nilai dari hasil tes kemampuan peserta seleksi bidang <i>Multimedia</i>
-----	--	---

Sumber: [Wawancara]

- Mahir PHP dan HTML

Nilai parameter ‘Mahir PHP dan HTML’ ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *programmer*. Parameter ini memiliki rentang nilai 0-17 yang menilai seberapa mahir mahasiswa dalam menggunakan bahasa pemrograman PHP dan HTML. Rentang nilai tersebut mewakili 17 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Menguasai Javascript dan CSS

Nilai parameter ‘Menguasai Javascript dan CSS’ didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *programmer*. Parameter ini memiliki rentang nilai 0-10 yang menilai seberapa jauh mahasiswa dalam menguasai bahasa pemrograman Javascript dan CSS. Rentang nilai tersebut mewakili 10 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Kemampuan MySQL dan SQL Server

Nilai parameter ‘Kemampuan MySQL dan SQL Server’ ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *programmer*. Parameter ini memiliki rentang nilai 0-4 yang menilai seberapa jauh kemampuan mahasiswa dalam menggunakan bahasa pemrograman SQL. Rentang nilai tersebut mewakili 4 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang

digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Menguasai HTML dan Javascript

Nilai parameter 'Kemampuan MySQL dan SQL Server' ini didapatkan dari hasil tes kemampuan peserta seleksi *web design*. Parameter ini memiliki rentang nilai 0-12 yang menilai seberapa jauh kemampuan mahasiswa dalam menggunakan bahasa pemrograman SQL. Rentang nilai tersebut mewakili 12 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Berpengalaman menggunakan Blender/PHP MySQL

Nilai parameter 'Berpengalaman menggunakan Blender/PHP MySQL' ini didapatkan dari hasil tes kemampuan peserta seleksi *web design*. Parameter ini memiliki rentang nilai 0-4 yang menilai seberapa jauh mahasiswa berpengalaman dalam menggunakan aplikasi Blender dan bahasa pemrograman PHP dan SQL. Rentang nilai tersebut mewakili 4 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Menguasai Adobe Photoshop, Dreamweaver, dan Illustrator

Nilai parameter 'Menguasai Adobe Photoshop, Dreamweaver, dan Illustrator' ini didapatkan dari hasil tes kemampuan peserta seleksi *web design*. Parameter ini memiliki rentang nilai 0-7 yang menilai seberapa jauh mahasiswa dalam menguasai penggunaan aplikasi Adobe Photoshop, Dreamweaver, dan Illustrator. Rentang nilai tersebut mewakili 7 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang

digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Menguasai *Network Component, Network OS, Basic Computer, Hardware Component*

Nilai parameter ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *network admin*. Parameter ini memiliki rentang nilai 0-5 yang menilai seberapa jauh kemampuan mahasiswa dalam menguasai komponen jaringan seperti *Network Component, Network OS, Basic Computer, Hardware Component*. Rentang nilai tersebut mewakili 5 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Memahami TCP/IP, LAN, WAN, Switches, Router *Configuration*

Nilai parameter ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *network admin*. Parameter ini memiliki rentang nilai 0-17 yang menilai seberapa jauh mahasiswa dalam memahami TCP/IP, LAN, WAN, Switches, Router *Configuration*. Rentang nilai tersebut mewakili 17 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Familiar dengan Windows dan Linux

Nilai parameter 'Familiar dengan Windows dan Linux' ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *network admin*. Parameter ini memiliki rentang nilai 0-5 yang menilai seberapa familiar mahasiswa dengan Sistem Operasi Windows dan Linux. Rentang nilai tersebut mewakili 5 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang

digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Familiar dengan IP Network & Fiber Network

Nilai parameter 'Familiar dengan IP Network & Fiber Network' ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *network admin*. Parameter ini memiliki rentang nilai 0-7 yang menilai seberapa familiar mahasiswa dengan IP Network & Fiber Network. Rentang nilai tersebut mewakili 7 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Pengalaman menggunakan *Popular Network Product*

Nilai parameter 'Pengalaman menggunakan *Popular Network Product*' ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang *network admin*. Parameter ini memiliki rentang nilai 0-100 yang menilai seberapa familiar mahasiswa dengan *Popular Network Product*. Rentang nilai tersebut mewakili 100 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Menguasai & memahami dasar Fotografi & Videografi

Nilai parameter ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang multimedia. Parameter ini memiliki rentang nilai 0-9 yang menilai seberapa jauh mahasiswa dalam menguasai dan memahami dasar fotografi dan videografi. Rentang nilai tersebut mewakili 9 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- Memahami teknik pembuatan skenario

Nilai parameter ‘Memahami teknik pembuatan skenario’ ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang multimedia. Parameter ini memiliki rentang nilai 0-4 yang menilai seberapa paham mahasiswa tentang pembuatan skenario. Rentang nilai tersebut mewakili 4 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- **Menguasai perangkat Audio Video Editing: Adobe Premiere, After Affect, DJ Studio, Sony**

Nilai parameter ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang multimedia. Parameter ini memiliki rentang nilai 0-5 yang menilai seberapa jauh mahasiswa menguasai perangkat Audio Video Editing seperti Adobe Premiere, After Effect, DJ Sony, Sony. Rentang nilai tersebut mewakili 5 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

- **Mampu berfikir kreatif & Mencipta ide-ide baru**

Nilai parameter ‘Mampu berfikir kreatif dan menciptakan ide-ide baru’ ini didapatkan dari hasil tes kemampuan peserta seleksi pada bidang multimedia. Parameter ini memiliki rentang nilai 0-4 yang menilai seberapa dalam kemampuan mahasiswa untuk berfikir kreatif dan menciptakan ide-ide baru. Rentang nilai tersebut mewakili 4 soal/pertanyaan yang memiliki 1 poin tambah untuk setiap soal/pertanyaan dengan jawaban benar. Apabila jawaban salah maka poin dari soal/pertanyaan tersebut sama dengan nol. Semua pertanyaan yang digunakan sebagai ukuran penilaian pada parameter ini ditunjukkan pada lampiran.

Dengan adanya sistem seleksi yang mengikut-sertakan beberapa tahap maka diharapkan mahasiswa yang lolos sebagai Student Employee dapat bekerja sesuai

dengan harapan guna membantu kegiatan pengembangan sistem yang ada di BPTIK UB.

2.3. Sistem Pendukung Keputusan

Sistem ini berbasis komputer yang terdiri dari tiga komponen yang saling berinteraksi yaitu sistem bahasa, sistem pengetahuan dan sistem pemrosesan masalah. Pembahasan dalam sub bab ini meliputi karakteristik dan kemampuan SPK serta komponen yang terdapat pada SPK.

2.3.1. Pengertian Pengambilan Keputusan dan Sistem Pendukung Keputusan

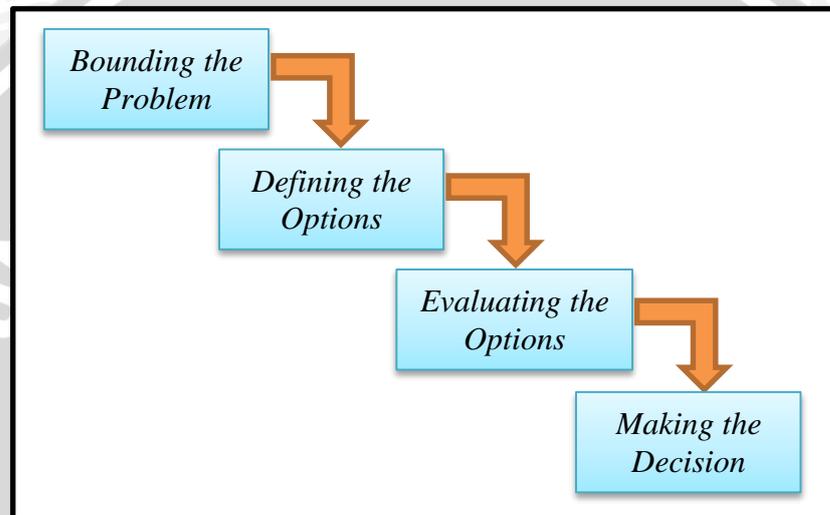
Pengambilan Keputusan adalah proses yang kompleks dan dipengaruhi oleh banyak faktor baik manusia dan non-manusia. Proses pengambilan keputusan menggunakan empat elemen *roadmap* sebagai panduan untuk proses pengambilan keputusan. Setiap kelompok dapat mengidentifikasi apa yang harus dilakukan nantinya dan juga dapat memperkirakan dimana terjadinya. *Roadmap* adalah satu set peluang dalam aliran logis, yang mengarah ke pengembangan dari sebuah keputusan yang kuat [POU-06].

Decision Support System atau Sistem Pendukung Keputusan (SPK) adalah sistem berbasis komputer yang menyatukan informasi dari berbagai sumber, membantu organisasi dan analisis informasi serta memfasilitasi evaluasi asumsi yang mendasari penggunaan model tertentu. SPK memungkinkan pembuat keputusan untuk mengakses data yang relevan di seluruh organisasi karena mereka membutuhkannya untuk membuat pilihan di antara beberapa alternatif. SPK memungkinkan pengambilan keputusan untuk menganalisa data yang dihasilkan dari sistem pemrosesan transaksi dan sumber informasi internal dengan mudah. SPK juga memungkinkan akses ke informasi eksternal dari organisasi serta memungkinkan pengambil keputusan untuk menganalisis informasi yang berperan dalam ketelitian keputusan dan akan memberikan dukungan yang interaktif [SAU-10].

SPK merupakan area pembuatan aplikasi sistem informasi, yang membantu para pembuat keputusan untuk menarik suatu keputusan yang efisien di suatu waktu. SPK menyediakan bantuan yang mudah dimengerti bagi para pembuat keputusan non teknis untuk dapat menemukan metode terbaik dengan cepat. SPK

adalah perangkat lunak yang menetapkan hubungan yang diperlukan antara kondisi saat ini dan kebutuhan manajemen yang diperlukan [POU-06].

Tahapan dalam pengambilan keputusan yaitu pembatasan masalah, definisi alternatif keputusan, dan membuat keputusan seperti yang ditunjukkan melalui Gambar 2.4 [POU-06]. *Bounding the problem* yaitu memberikan batasan terhadap masalah, *defining the option* yaitu mendefinisikan alternatif keputusan, *evaluating the option* yaitu mengevaluasi alternatif keputusan, dan yang terakhir *making the decision* yaitu membuat keputusan.



Gambar 2.4. Empat elemen *roadmap* untuk mencapai pengambilan keputusan yang baik.

Sumber:[POU-06]

2.3.2. Karakteristik Sistem Pendukung Keputusan

Terdapat 3 karakter pada sistem pendukung keputusan yaitu karakteristik pertama, SPK harus mengakses data dari berbagai sumber. Karakteristik kedua adalah SPK memfasilitasi pengembangan dan evaluasi model dari proses pemilihan. Artinya, SPK harus memungkinkan pengguna untuk mengubah sejumlah besar “data” menjadi “informasi” yang membantu mereka membuat keputusan yang baik. Model dapat berupa *summarization* sederhana atau mungkin model matematika canggih. Karakteristik ketiga yaitu SPK harus menyediakan antarmuka pengguna yang baik dimana pengguna dapat dengan mudah melakukan navigasi dan interaksi [SAU-10].

2.3.3. Komponen Sistem Pendukung Keputusan

Ada 3 komponen Sistem Pendukung Keputusan yaitu [SAU-10]:

1. *Data Component* (Manajemen Data)

Database Management System (DBMS) menyediakan akses ke data serta semua program kontrol yang diperlukan untuk mendapatkan data tersebut dalam bentuk yang sesuai untuk analisis dalam pertimbangan. Data mencakup fakta tentang operasi internal, tren, kecerdasan dan atau riset pasar, dan informasi yang tersedia secara umum. DBMS harus cukup canggih untuk memberikan akses pengguna ke data bahkan ketika mereka tidak tahu dimana data berada secara fisik. DBMS memfasilitasi penggabungan data dari sumber yang berbeda. DBMS juga harus cukup canggih untuk menggabungkan data tanpa instruksi yang jelas dari pengguna mengenai bagaimana seseorang menyelesaikan tugas itu.

2. *Model Management* (Manajemen Model)

Model Base Management System (MBMS) melacak semua model dalam SPK yang mungkin dijalankan selama analisis serta kontrol untuk menjalankan model. MBMS mungkin terdiri dari sintak yang diperlukan untuk menjalankan pekerjaan, format dimana data harus dimasukkan sebelum menjalankan model (dan untuk menempatkan data seperti dalam format), dan format data setelah menjalankan pekerjaan. MBMS juga menghubungkan antara model sehingga *output* dari suatu model dapat menjadi *input* ke dalam model lain. MBMS menyediakan mekanisme untuk analisis sensitifitas dari model setelah dijalankan. MBMS memberikan bantuan konteks-sensitif dan model yang sensitif untuk membantu asumsi model pertanyaan pengguna untuk menentukan apakah mereka sudah sesuai untuk keputusan dalam pertimbangan.

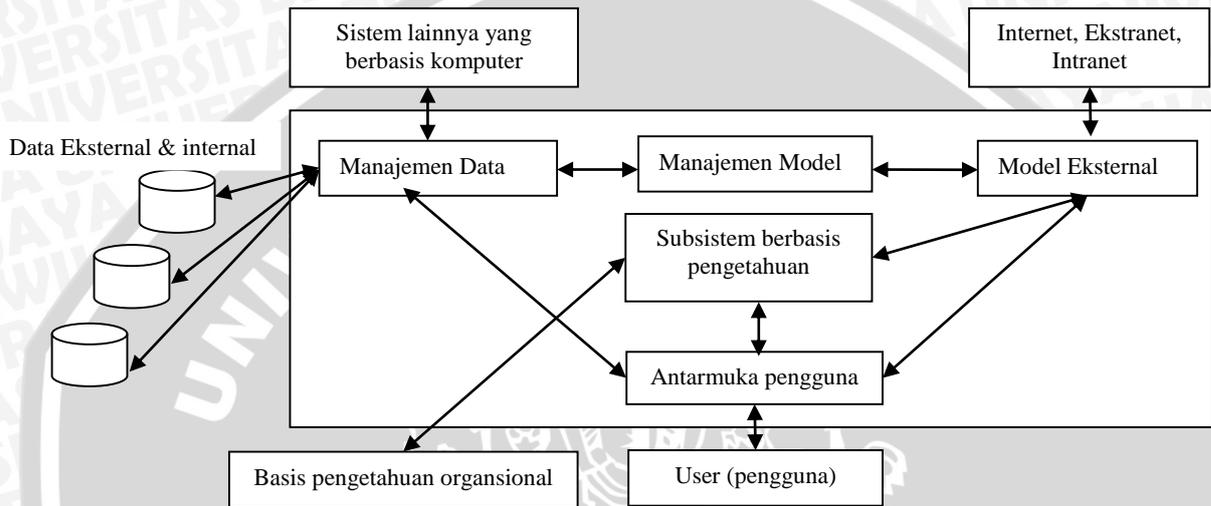
3. *User Interface* (Antarmuka Pengguna)

User Interface atau antar muka pengguna merupakan semua mekanisme dimana informasi adalah masukan ke sistem dan keluaran dari sistem. *User Interface* terdiri dari semua layar *input* oleh pengguna yang meminta data dan model. *User Interface* juga terdiri dari semua layar *output* dimana pengguna mendapatkan hasil. Banyak pengguna berpikir antarmuka pengguna sebagai SPK yang nyata karena itu adalah bagian dari sistem yang mereka lihat.

Ada satu lagi komponen SPK yaitu manajemen berbasis pengetahuan yang sifatnya opsional, tetapi bisa memberikan banyak manfaat karena memberikan intelegensi bagi ketiga komponen utama tersebut. Komponen manajemen berbasis

pengetahuan juga bisa diinterkoneksi dengan repositori pengetahuan perusahaan (bagian dari sistem manajemen pengetahuan), yang kadang-kadang disebut basis pengetahuan organisasional [KUS-07].

Adapun arsitektur sistem pendukung keputusan sebagai gambaran komponen-komponen yang telah dijelaskan di atas ditunjukkan melalui Gambar 2.5.



Gambar 2.5. Arsitektur Sistem Pendukung Keputusan

Sumber: [[KUS-07]]

2.4. Support Vector Machine (SVM)

SVM dapat dikatakan sebagai metode pengklasifikasian yang akurat. Metode klasifikasi SVM didasarkan pada prinsip *Structural Risk Minimization* dari teori pembelajaran komputasional. SVM membutuhkan *training* set positif dan negatif. *Training* set positif dan negatif ini dibutuhkan SVM untuk membuat keputusan terbaik dalam memisahkan data positif dengan data negatif di ruang n -dimensi, yang disebut dengan *hyperplane*. Jadi, SVM adalah metode klasifikasi *supervised learning* untuk mencari garis pemisah *hyperplane* dengan mengoptimalkan *hyperplane*, dan memaksimalkan *margin* antara dua kelas [KHA-10].

Pada permasalahan *nonlinear* SVM menggunakan sebuah pemetaan *nonlinear* yang disebut fungsi kernel untuk mengubah data training asli ke dalam ruang fitur (*feature space*) berdimensi tinggi. Penelitian ini akan dilakukan secara *nonlinear* proses klasifikasi. Fungsi pemisah yang akan dipisahkan oleh bidang pembatas berupa suatu fungsi *nonlinear*.

2.4.1. SVM Linier

Pada kasus klasifikasi dengan SVM secara linier. Misalkan $x_i \in \{x_1, \dots, x_n\}$ adalah dataset dan $y_i \in \{+1, -1\}$ adalah label kelas dari data x_i . Hal pertama yang harus dilakukan adalah dengan mencari *hyperplane* dengan fungsi $f(w, b) = x_i \cdot w + b$ yang dipisahkan menjadi kelompok positif dan negatif. Persamaan untuk mencari *hyperplane* tersebut ditunjukkan pada persamaan (2-1) dan (2-2) [VIJ-99].

$$x_i \cdot w + b \geq 1 \text{ untuk } y_i = 1 \dots\dots\dots (2-1)$$

$$x_i \cdot w + b \leq -1 \text{ untuk } y_i = -1 \dots\dots\dots (2-2)$$

keterangan:

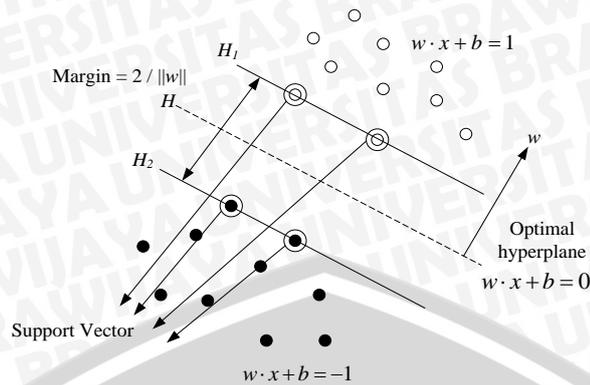
- x_i = data ke- i
- w = vektor yang tegak lurus terhadap *hyperplane*
- b = nilai bias (*threshold*)
- y_i = kelas data ke- i

yang ekuivalen dengan persamaan (2-3) [VIJ-99].

$$y_i(x_i \cdot w + b) - 1 \geq 0 \text{ untuk } i = 1, \dots, n. \dots\dots\dots (2-3)$$

dimana: n = jumlah data

Hyperplane terbaik adalah *hyperplane* yang terletak di tengah-tengah antara dua set objek dari dua kelas. Mencari *hyperplane* terbaik ekuivalen dengan memaksimalkan margin atau jarak antara dua set obyek dari kelas yang berbeda. Jika $w x_1 + b = +1$ adalah *hyperplane* pendukung (*supporting hyperplane*) dari kelas $+1$ atau bisa disebut juga sebagai H_1 dan $w x_2 + b = -1$ merupakan *hyperplane* pendukung dari kelas -1 atau bisa juga disebut sebagai H_2 , maka margin antara kedua kelas tersebut dapat dihitung dengan mencari jarak antara kedua *hyperplane*-pendukung dari kedua kelas tersebut. Ilustrasi SVM dengan *optimal hyperplane* dapat dilihat pada Gambar 2.6 [SAN-09].



Gambar 2.6. Ilustrasi Optimal *Hyperplane*
 Sumber: [SAN-09]

$$\text{Margin} = |d_{H_1} - d_{H_2}| = \frac{2}{\|w\|} \dots\dots\dots (2-4)$$

dimana:

d_{H_1} = jarak *hyperplane* pendukung kelas +1

d_{H_2} = jarak *hyperplane* pendukung kelas -1

w = vektor yang tegak lurus terhadap *hyperplane*

Selanjutnya untuk menentukan *hyperplane* pemisah kedua kelas, maka margin perlu dimaksimalkan menggunakan persamaan (2-5) [VIJ-99]:

$$\text{Minimize } J_1[w] = \frac{1}{2} \|w\|^2 \dots\dots\dots (2-5)$$

$$\text{dengan } y_i(x_i \cdot w + b) - 1 \geq 0, i = 1, \dots, l \dots\dots\dots (2-6)$$

dimana:

x_i = data ke- i

y_i = kelas data ke- i

b = nilai bias (*threshold*)

w = vektor yang tegak lurus terhadap *hyperplane*

2.4.2. SVM Non Linier

Pada beberapa kasus, data tidak bisa diklasifikasi menggunakan metode *linier* SVM, sehingga perlu digunakan variabel *slack* untuk menangani *misclassification* atau kesalahan klasifikasi. Variabel *slack* bertujuan untuk mengatasi kondisi ketidaklayakan (*infeasibility*) seperti ditunjukkan pada pertidaksamaan (2-7) [VIJ-99].

$$y_i(x_i \cdot w + b) - 1 + \xi_i \geq 0 \text{ untuk } i = 1, \dots, n \dots\dots\dots (2-7)$$

Maka untuk memaksimalkan margin dan mengurangi jumlah *misclassification* digunakanlah persamaan (2-8) [VIJ-99].

$$\text{Minimize } J_2[w, \xi_i] = \left(\frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i\right) \dots\dots\dots (2-8)$$

$$\text{dengan } y_i(x_i \cdot w + b) - 1 + \xi_i \geq 0, \dots\dots\dots (2-9)$$

$$\xi_i \geq 0, \quad i = 1, \dots, n \dots\dots\dots (2-10)$$

keterangan:

x_i = data ke- i

y_i = kelas data ke- i

b = nilai bias (*threshold*)

ξ_i = variabel *slack* (mengukur *error* dari data)

w = bobot *support vector*

C = parameter *user* bernilai bilangan positif (batasan *error*)

n = jumlah data

J_2 = fungsi *minimize*

Nilai C (*Complexity*) bertujuan untuk meminimalkan *error* dan memperkecil nilai variabel *slack*. Nilai C yang besar berarti memberikan penalti yang lebih besar terhadap *error* klasifikasi [NUG-03]. Bisa juga dituliskan menggunakan *Lagrange multiplier* yang ditunjukkan pada persamaan (2-11) [VIJ-99].

$$\text{Minimize } L(w, b, \xi_i, h, r) = \frac{1}{2} \|w\|^2 + C \sum_{i=1}^l \xi_i - \sum_{i=1}^l h_i [y_i(x_i \cdot w + b) - 1 + \xi_i] - \sum_{i=1}^l r_i \xi_i \dots\dots\dots (2-11)$$

dimana $h = (h_1, \dots, h_l)$ dan $r = (r_1, \dots, r_l)$ adalah *non-negative Lagrange multiplier*.

Persamaan (2-11) dapat diubah menjadi *dual problem* yang lebih mudah ditangani yang ditunjukkan melalui persamaan (2-12) [VIJ-99].

$$\text{Maximize } L_D(h) = \left(\sum_{i=1}^l h_i - \frac{1}{2} h \cdot Dh\right) \dots\dots\dots (2-12)$$

$$\text{dengan } \sum_{i=1}^l y_i h_i = 0, \dots\dots\dots (2-13)$$

$$0 \leq h_i \leq C, \quad i = 1, \dots, n \dots\dots\dots (2-14)$$

dimana $D'_{ij} = y_i y_j x_i \cdot x + \lambda^2 y_i y_j$

Penyelesaian *dual problem* tersebut biasanya menggunakan *quadratic programming*. Selanjutnya setelah *optimal multiplier* h_i ditemukan, maka klasifikasinya adalah dengan menggunakan fungsi yang ditunjukkan pada persamaan (2-15) [VIJ-99].

$$f(x) = \sum_i h_i y_i x_i \dots x + b \dots \dots \dots (2-15)$$

Nilai bias b dapat ditemukan menggunakan *support vector* positif dan negatif, x^+ dan x^- seperti yang ditunjukkan pada persamaan (2-16) dan (2-17) [BOS-02].

$$(w \cdot x^+ + b) = +1) \dots \dots \dots (2-16)$$

$$(w \cdot x^- + b) = -1) \dots \dots \dots (2-17)$$

dimana w dinyatakan seperti pada persamaan (2-18) [BOS-02].

$$w = \sum_{i=1}^n \alpha_i y_i x_i \dots \dots \dots (2-18)$$

Keterangan:

$\alpha_i = h_i =$ Lagrange multiplier data ke-i

$y_i =$ kelas data ke-i

$x_i =$ data ke-i

$x^+ =$ data kelas positif yang memiliki nilai *Lagrange multiplier* terbesar

Sehingga dapat dinyatakan persamaan nilai bias b sebagai berikut [BOS-02]:

$$b = -\frac{1}{2}(w \cdot x^+ + w \cdot x^-) \dots \dots \dots (2-19)$$

Karena keterbatasan SVM, maka kemudian dikembangkan fungsi kernel untuk mengklasifikasikan data dengan bentuk *non linier*. Dalam *non linear SVM*, pertama-tama data x dipetakan oleh fungsi $\Phi(x)$ ke ruang vektor yang berdimensi lebih tinggi. Pada ruang vektor yang baru ini, hyperplane yang memisahkan kedua *class* tersebut dapat dikonstruksikan. Hal ini sejalan dengan teori Cover yang menyatakan “*Jika suatu transformasi bersifat non linear dan dimensi dari feature space cukup tinggi, maka data pada input space dapat dipetakan ke feature space yang baru, dimana pattern-pattern tersebut pada probabilitas tinggi dapat dipisahkan secara linear*”[NUG-03].

Pemetaan ini dilakukan dengan menjaga topologi data, dalam artian dua data yang berjarak dekat pada *input space* akan berjarak dekat juga pada *feature space*, sebaliknya dua data yang berjarak jauh pada *input space* akan juga berjarak



jauh pada *feature space*. Selanjutnya proses pembelajaran pada SVM dalam menemukan titik-titik support vector, hanya bergantung pada *dot product* dari data yang sudah ditransformasikan pada ruang baru yang berdimensi lebih tinggi, yaitu $\Phi(x_i) \cdot \Phi(x_j)$ [NUG-03]. Karena umumnya transformasi Φ ini tidak diketahui, dan sangat sulit untuk difahami secara mudah, maka perhitungan *dot product* tersebut sesuai teori Mercer dapat digantikan dengan fungsi kernel $K(x_i, x_j)$ yang mendefinisikan secara implisit transformasi Φ . Hal ini disebut sebagai *Kernel Trick*, yang dirumuskan seperti persamaan (2-20) [NUG-03].

$$K(x_i, x_j) = \Phi(x_i) \cdot \Phi(x_j) \dots\dots\dots (2-20)$$

Kernel trick memberikan berbagai kemudahan, karena dalam proses pembelajaran SVM, untuk menentukan *support vector*, kita hanya cukup mengetahui fungsi kernel yang dipakai, dan tidak perlu mengetahui wujud dari fungsi *non linear* Φ . Beberapa macam fungsi kernel SVM dapat dilihat pada Tabel 2.4.

Tabel 2.4. Berbagai Fungsi Kernel

No	Nama Kernel	Definisi Fungsi
1	<i>Linier</i>	$K(x, y) = x \cdot y$
2	<i>Polynomial of degree d</i>	$K(x, y) = (x \cdot y)^d$
3	<i>Polynomial of degree up to d</i>	$K(x, y) = (x \cdot y + c)^d$
4	<i>Gaussian RBF</i>	$K(x, y) = \exp\left(\frac{-\ x - y\ ^2}{2\sigma^2}\right)$
5	<i>Sigmoid (Tangen Hiperbolik)</i>	$K(x, y) = \tanh(\sigma(x \cdot y) + c)$
6	Invers Multi Kuadratik	$K(x, y) = \frac{1}{\sqrt{\ x - y\ ^2 + c^2}}$
7	<i>Additive</i>	$K(x, y) = \sum_{i=1}^n K_i(x, y)$

Sumber: [CHO-13]

Selanjutnya hasil klasifikasi dari data x diperoleh dari persamaan berikut [NUG-03]:

$$f(\Phi(x)) = w \cdot \Phi(x) + b \dots\dots\dots (2-21)$$



$$= \sum_{i=1}^n \alpha_i y_i \Phi(x_i) \cdot \Phi(x) + b \dots\dots\dots (2-22)$$

$$= \sum_{i=1}^n \alpha_i y_i K(x_i, x) + b \dots\dots\dots (2-23)$$

dimana,

x_i = data ke-i

α_i = Lagrange multiplier data ke-i

y_i = kelas data ke-i

w = bobot vektor

2.4.3. Sequential Training SVM

Pada penggunaan metode SVM dalam mencari *hyperplane* yang optimal digunakanlah *quadratic programming* (QP). Namun penyelesaian QP biasanya cukup kompleks, membuang-buang waktu, dan rentan terhadap ketidakstabilan numerik. Kemudian dikembangkan metode sequential training SVM untuk klasifikasi, algoritmanya adalah sebagai berikut [VIJ-99]:

1. Menginisialisasi $\alpha_i = 0$ dan parameter lain, misalnya $\lambda = 2$, $\gamma = 2$, $C = 1$, IterMax=100, dan $\varepsilon = 0.0001$. Kemudian hitung matriks Hessian seperti yang ditunjukkan melalui persamaan (2-17).

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2) \dots\dots\dots (2-17)$$

untuk $i, j = 1, \dots, n$.

keterangan:

x_i = data ke-i

x_j = data ke-j

y_i = kelas data ke-i

y_j = kelas data ke-j

n = jumlah data

λ = batas teoritis yang akan diturunkan

$K(x_i, x_j)$ = fungsi kernel yang digunakan

2. Mulai dari data ke i sampai l , hitung menggunakan persamaan berikut:

- a.
$$E_i = \sum_{j=1}^l h_j D_{ij}$$



$$b. \quad \delta\alpha_i = \min\{\max[\gamma(1-E_i), -\alpha_i], C-\alpha_i\}$$

$$c. \quad \alpha_i = \alpha_i + \delta\alpha_i$$

dimana:

α_i = *Langrange multiplier*, data dengan $\alpha_i > 0$ disebut *support vector*

C (*Complexity*) = parameter *user* bernilai bilangan positif (batasan *error*)

E_i = *Error* data ke- i

γ = parameter untuk mengontrol kecepatan proses *learning*

Fungsi $\delta\alpha_i$ adalah fungsi konvergensi untuk memantau perubahan fungsi *Lagrange multiplier*.

3. Langkah 2 dilakukan terus-menerus hingga kondisi iterasi maksimum tercapai atau nilai α mencapai nilai konvergen dimana $\max(|\delta\alpha_i|) < \varepsilon$.
4. Selanjutnya didapatkan nilai *support vector* (SV), $SV = (\alpha_i > \text{ThresholdSV})$. Nilai *ThresholdSV* didapatkan dari beberapa percobaan, Biasanya digunakan $\text{ThresholdSV} \geq 0$

2.5. Analytical Hierarchy Process (AHP)

Metode AHP dikembangkan oleh Thomas L. Saaty, seorang ahli matematika. Metode ini adalah kerangka untuk mengambil keputusan dengan efektif atas persoalan yang kompleks dengan menyederhanakan dan mempercepat proses pengambilan keputusan dengan memecahkan persoalan tersebut kedalam bagian-bagiannya, menata bagian atau variabel dan mensintesis berbagai pertimbangan ini untuk mempengaruhi hasil pada situasi tersebut [IDA-11].

Metode AHP ini membantu memecahkan persoalan yang kompleks dengan menstruktur suatu hirarki kriteria, pihak yang berkepentingan, hasil dan dengan menarik berbagai pertimbangan guna mengembangkan bobot atau prioritas. Metode ini juga menggabungkan kekuatan dari perasaan dan logika yang bersangkutan pada berbagai persoalan, lalu mensintesis berbagai pertimbangan yang beragam menjadi hasil yang cocok dengan perkiraan kita secara intuitif sebagaimana yang dipresentasikan pada pertimbangan yang telah dibuat [IDA-11].

Ada tiga prinsip dalam memecahkan persoalan dengan AHP, yaitu prinsip menyusun hirarki (*Decomposition*), prinsip menentukan prioritas (*Comparative*

Judgement) dan prinsip konsisten logis (*Logical Consistency*). Hirarki yang dimaksud adalah hirarki dari permasalahan yang akan dipecahkan untuk mempertimbangkan kriteria-kriteria atau komponen-komponen yang mendukung pencapaian tujuan. Database adalah kumpulan dari elemen data yang saling berhubungan [IDA-11].

Tahapan-tahapan proses dalam metode AHP [ANS-12]:

- 1) Mendefinisikan masalah dan menentukan tujuan yang diinginkan.
- 2) Membuat struktur hirarki yang diawali dengan tujuan umum, dilanjutkan dengan kriteria-kriteria dan alternatif-alternatif pilihan.
- 3) Membentuk matriks perbandingan berpasangan yang menggambarkan pengaruh setiap elemen terhadap masing-masing kriteria.
- 4) Menguji konsistensi hirarki. Jika nilai memenuhi standar yang ditetapkan yaitu *Consistency Ratio* (CR) $< 0,1$ maka penilaian harus diulang kembali.

2.5.1. Kelebihan AHP

Kelebihan dari model AHP dibandingkan dengan model pengambilan keputusan yang lain terletak pada kemampuannya untuk memecahkan masalah yang multiobjectives dengan multikriteria. Kebanyakan model yang sudah ada memakai single objectives dengan multikriteria. Model *Linear Programming* misalnya, memakai satu tujuan dengan banyak kendala (kriteria). Kelebihan model AHP ini lebih disebabkan oleh fleksibilitasnya yang tinggi terutama dalam pembuatan hirarki. Sifat fleksibel tersebut membuat model AHP dapat menangkap beberapa tujuan dan beberapa kriteria sekaligus dalam sebuah model atau sebuah hirarki [IDA-11].

2.5.2. Kekurangan AHP

Disamping kelebihan-kelebihan yang dimiliki, model AHP juga mempunyai beberapa kelemahan. Ketergantungan model ini terhadap input berupa persepsi seorang ahli akan membuat hasil akhir dari model ini menjadi tidak ada artinya apabila si ahli memberikan penilaian yang keliru. Kebanyakan orang bertanya apakah persepsi dari seorang ahli tersebut dapat mewakili kepentingan orang banyak atau tidak. Keraguan seperti ini tidak lain disebabkan oleh kenyataan bahwa setiap orang mempunyai persepsi yang berbeda dengan orang lain. Karenanya, untuk model AHP ini dapat diterima oleh masyarakat, perlu diberikan

kriteria dan batasan tegas dari seorang ahli serta meyakinkan masyarakat untuk menganggap bahwa persepsi si ahli dapat mewakili pendapat masyarakat atau paling tidak sebagian masyarakat [IDA-11].

2.5.3. Prinsip Dasar AHP

Dalam menyelesaikan persoalan dengan metode AHP, ada beberapa prinsip dasar yang harus dipahami yaitu [ANS-12]:

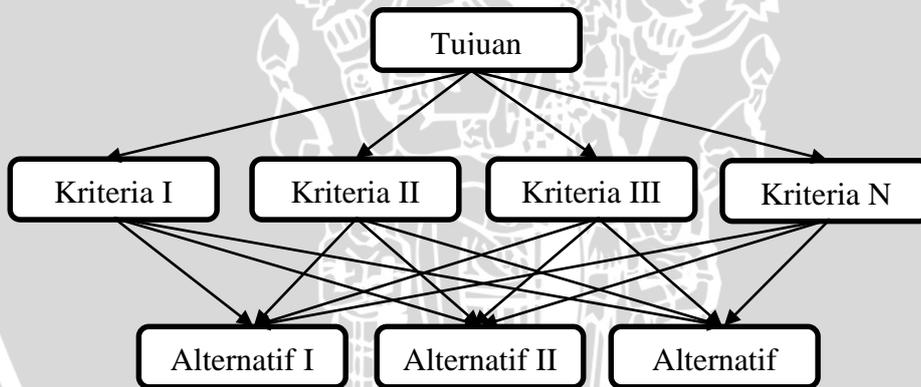
a. *Decomposition.*

Decomposition adalah langkah memecahkan atau membagi masalah yang utuh menjadi elemen-elemen ke bentuk hirarki, dimana setiap elemen saling berhubungan.

Bentuk struktur dekomposisi yaitu [ANS-12]:

- Tingkat pertama : Tujuan keputusan (Goal)
- Tingkat kedua : Kriteria-kriteria
- Tingkat ketiga : Alternatif-alternatif

Dekomposisi masalah dapat digambarkan seperti Gambar 2.7.



Gambar 2.7. Struktur Dekomposisi
Sumber: [ANS-12]

b. *Comparative Judgement*

Comparative judgement dilakukan dengan memberikan penilaian tentang kepentingan relatif antar kriteria. Hasil dari penilaian ini disajikan dalam bentuk matriks perbandingan berpasangan atau matriks keputusan.

c. *Synthesis of Priority*

Dari matriks keputusan yang terbentuk dapat ditentukan nilai bobot untuk masing-masing kriteria sehingga bisa didapatkan prioritas antar kriteria.

2.5.4. Penyusunan Prioritas

Setiap elemen yang terdapat dalam hirarki harus diketahui bobot relatifnya satu sama lain. Tujuannya adalah untuk mengetahui tingkat kepentingan dalam permasalahan terhadap kriteria dan struktur hirarki atau permasalahan secara keseluruhan.

Langkah pertama dilakukan dalam menentukan prioritas kriteria adalah menyusun perbandingan berpasangan, yaitu membandingkan dalam bentuk berpasangan seluruh kriteria untuk setiap sub sistem hirarki. Perbandingan tersebut kemudian ditransformasikan kedalam bentuk matriks perbandingan berpasangan untuk analisis numerik. Nilai numerik yang di berikan untuk seluruh perbandingan diperoleh dari skala perbandingan dari 1 - 9 yang telah ditetapkan oleh Saaty, seperti pada Tabel 2.5 [ANS-12].

Tabel 2.5. Skala Penilaian Perbandingan Berpasangan

Skala	Pasangan	Definisi
1	1	Sama pentingnya
3	1/3	Agak lebih penting yang satu atas yang lain
5	1/5	Cukup penting
7	1/7	Sangat penting
9	1/9	Mutlak lebih penting
2, 4, 6, 8	1/2, 1/4, 1/6, 1/8	Nilai tengah

Sumber: [ANS-12]

2.5.5. Matriks Keputusan

Apabila pengambil keputusan sudah memasukkan persepsinya atau penilaian untuk setiap perbandingan antara kriteria-kriteria yang berada dalam satu level (tingkatan) atau yang dapat diperbandingkan maka untuk mengetahui kriteria mana yang paling disukai atau paling penting, disusun sebuah matriks perbandingan disetiap level (tingkatan) [ANS-12].

2.5.6. Uji Konsistensi dan Indeks Rasio

Dengan metode AHP yang memakai persepsi pembuat keputusan sebagai inputnya maka ketidakkonsistenan mungkin terjadi karena manusia memiliki keterbatasan dalam menyatakan persepsinya secara konsisten terutama kalau harus

membandingkan banyak kriteria. Berdasarkan kondisi ini maka pembuat keputusan dapat menyatakan persepsinya tersebut akan konsisten atau tidak.

Thomas L. Saaty telah membuktikan bahwa indeks konsistensi dari matriks berordo dapat diperoleh melalui persamaan (2-21) [ANS-12]:

$$CI = \frac{(t-n)}{(n-1)} \dots\dots\dots (2-21)$$

CI = Consistency Index

t = Nilai normalisasi terbesar dari matriks berordo n atau dapat dinyatakan dengan notasi λ_{maks}

n = Jumlah objek (kriteria/alternatif)

Apabila bernilai nol, maka matriks *pair-wise comparison* tersebut konsisten (Thomas L.Saaty, 2008). Batas ketidakkonsistenan (*inconsistency*) yang telah ditetapkan oleh Thomas L.Saaty ditentukan dengan menggunakan persamaan Rasio Konsistensi (*Consistency Ratio* = CR), yaitu perbandingan indeks konsistensi dengan nilai *Random Indeks* (RI) yang didapatkan dari suatu eksperimen oleh Oak Ridge National Laboratory kemudian dikembangkan oleh Wharton School dan diperlihatkan seperti pada Tabel 2.6 [ANS-12].

Tabel 2.6. Random Indeks (RI)

n	RI
1	0
2	0
3	0,58
4	0,90
5	1,12
6	1,24
7	1,32
8	1,41
9	1,45
10	1,49
11	1,51
12	1,48
13	1,56

14	1,57
15	1,58

Sumber: [ANS-12]:

Dengan persamaan Rasio Konsistensi ditunjukkan melalui persamaan (2-22) [ANS-12]:

$$CR = \frac{CI}{RI} \dots\dots\dots (2-22)$$

CR = Rasio Konsistensi (*Consistency Ratio*)

RI = Indeks Random (*Random Index*)

Bila matriks *pair-wise comparison* dengan nilai $\leq 0,1$ maka ketidak konsistenan pendapat dari pengambil keputusan masih dapat diterima dan jika tidak maka penilaian perlu diulang.

2.5.7. Perbandingan Berpasangan (*Pairwise Comparison*)

Pada model AHP pengambilan keputusan menentukan nilai atau skor tiap alternative untuk kriteria menggunakan Perbandingan Pasangan (*Pairwise Comparison*). Pada perbandingan pasangan, pembuatan keputusan memebandingkan dua alternatif berdasarkan suatu kriteria tertentu dan mengindikasikan suatu preferensi [IDA-11].

Standar skala preferensi yang digunakan AHP (untuk memasukkan nilai perbandingan berpasangan) diperlihatkan pada table dibawah. Suatu skala preferensi memeberikan nilai numeric untuk berbagai tingkat preferensi. Tiap tingkat pada skala dibuat berdasarkan perbandingan dua item.

2.5.8. Penggunaan Metode AHP

Analityc Hierarchy Process digunakan dalam memecahkan masalah diantaranya untuk mengalokasikan sumber daya, analisis keputusan manfaat atau biaya, menentukan peringkat beberapa alternatif, melaksanakan perancangan ke masa depan yang diproyeksikan dan menetapkan prioritas pengembangan suatu unit usaha dan permasalahan kompleks lainnya. Secara umum, langkah-langkah dasar dari AHP dapat diringkas dalam penjelasan berikut ini [IDA-11]:

1. Mendefinisikan masalah dan menetapkan tujuan. Bila AHP digunakan untuk memilih alternatif atau penyusunan prioritas alternatif maka pada tahap ini dilakukan pengembangan alternatif.
2. Menyusun masalah dalam struktur hirarki. Setiap permasalahan yang kompleks dapat ditinjau dari yang detail dan terstruktur.
3. Menyusun prioritas untuk tiap elemen masalah pada tiap hirarki. Proses ini menghasilkan bobot elemen terhadap pencapaian tujuan, sehingga elemen dengan bobot tertinggi memiliki prioritas paling tinggi.
4. Melakukan pengujian konsistensi terhadap perbandingan antar elemen yang didapatkan pada tiap tingkat hirarki.

Melakukan pengujian konsistensi pada hirarki. Pengujian bertujuan untuk menguji konsistensi perbandingan antara kriteria yang dilakukan untuk seluruh hirarki.

2.6. Fuzzy Analytical Hierarchy Process (FAHP)

Fuzzy AHP merupakan gabungan metode AHP dengan pendekatan konsep *fuzzy*. F-AHP menutupi kelemahan yang terdapat pada AHP yaitu permasalahan terhadap kriteria yang memiliki sifat subjektif lebih banyak. Ketidakpastian bilangan direpresentasikan dengan urutan skala.

2.6.1. Logika Fuzzy

Teori himpunan *fuzzy* diperkenalkan pertama kali oleh Lotfi A. Zadeh pada tahun 1965. Dalam logika *fuzzy* dikenal keadaan dari nilai “0” sampai ke nilai “1”. Logika *fuzzy* tidak hanya mengenal dua keadaan tetapi juga mengenal sejumlah keadaan yang berkisar dari keadaan salah sampai keadaan benar [ANS-12].

2.6.2. Himpunan Klasik

Pada himpunan tegas (*crisp*), nilai keanggotaan suatu item x dalam suatu himpunan A , yang sering ditulis dengan $\mu_A[x]$, memiliki 2 kemungkinan, yaitu [ANS-12]:

- Satu (1), yang berarti bahwa suatu item menjadi anggota dalam suatu himpunan, atau
- Nol (0), yang berarti bahwa suatu item tidak menjadi anggota dalam suatu himpunan.

2.6.3. Himpunan Fuzzy

Pada logika boolean, sebuah individu dipastikan sebagai anggota dari salah satu himpunan saja, sedangkan pada himpunan *fuzzy* sebuah individu dapat masuk pada dua himpunan yang berbeda. Seberapa besar eksistensinya dalam himpunan tersebut dapat dilihat pada nilai keanggotaannya [ANS-12].

Himpunan *fuzzy* A pada semesta X dinyatakan sebagai himpunan pasangan berurutan (*set of ordered pairs*) baik diskrit maupun kontinu seperti yang ditunjukkan pada persamaan (2-23) [ANS-12].

$$A = \{(x, \mu_A(x)) \mid x \in X\} \dots \dots \dots (2-23)$$

Dimana $\mu_A(x)$ adalah fungsi keanggotaan himpunan *fuzzy* A . Fungsi keanggotaan memetakan setiap $x \in X$ pada suatu nilai antara $[0,1]$ yang disebut derajat keanggotaan (*membership grade* atau *membership value*).

- Fuzzifikasi

Berfungsi untuk mengubah masukan yang bersifat *crisp* (bukan *fuzzy*) ke himpunan *fuzzy* dengan menggunakan aturan fuzzifikasi.

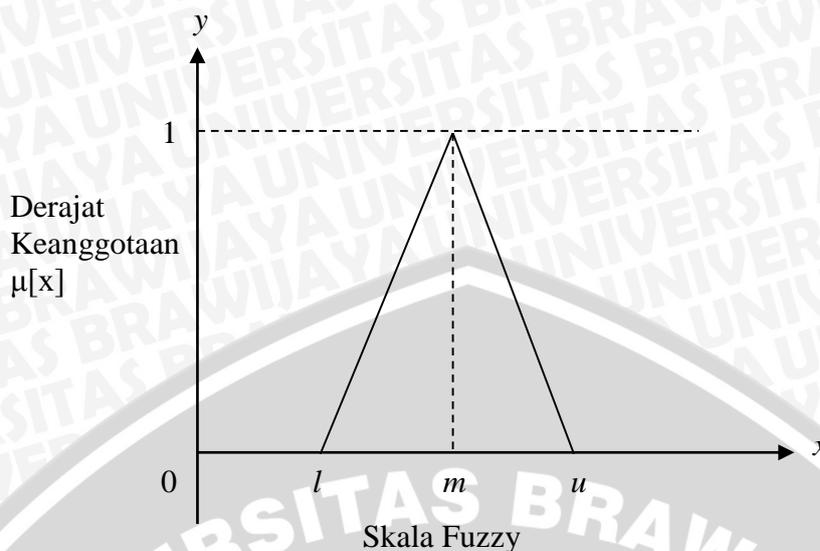
- Defuzzifikasi

Berfungsi untuk mentransformasikan bilangan-bilangan *fuzzy* (*fuzzy set*) yang bersifat *fuzzy* menjadi bentuk sebenarnya yang bersifat *crisp* dengan menggunakan aturan defuzzifikasi.

Pendekatan *triangular fuzzy number* dalam metode AHP adalah pendekatan yang digunakan untuk meminimalisasi ketidakpastian dalam skala AHP yang berbentuk nilai *crisp* (Deng, 1999). Cara pendekatan yang dilakukan adalah dengan melakukan fuzzifikasi pada skala AHP sehingga diperoleh skala baru yang disebut skala *fuzzy* AHP [ANS-12].

2.6.4. Derajat Keanggotaan dan Skala *Fuzzy* Segitiga

Penentuan derajat keanggotaan F-AHP yang dikembangkan oleh Chang (1996) menggunakan fungsi keanggotaan segitiga (*Triangular Fuzzy Number/TFN*) [JAS-11]. Fungsi keanggotaan segitiga merupakan gabungan antara dua garis (linear). Grafik fungsi keanggotaan segitiga digambarkan dalam bentuk kurva segitiga seperti terlihat pada Gambar 2.8.



Gambar 2.8. Fungsi Keanggotaan Segitiga

Sumber: [JAS-11]

Penelitian Jasril mendefinisikan nilai intensitas AHP ke dalam skala *fuzzy* segitiga yaitu membagi tiap himpunan *fuzzy* dengan dua (2), kecuali untuk intensitas kepentingan satu (1) [JAS-11].

Pada model AHP orisinil, *pairwise comparison* menggunakan skala 1 – 9 [ANS-12]. Dengan mentransformasi *Triangular Fuzzy Number* terhadap skala AHP maka skala yang digunakan adalah seperti Tabel 2.7.

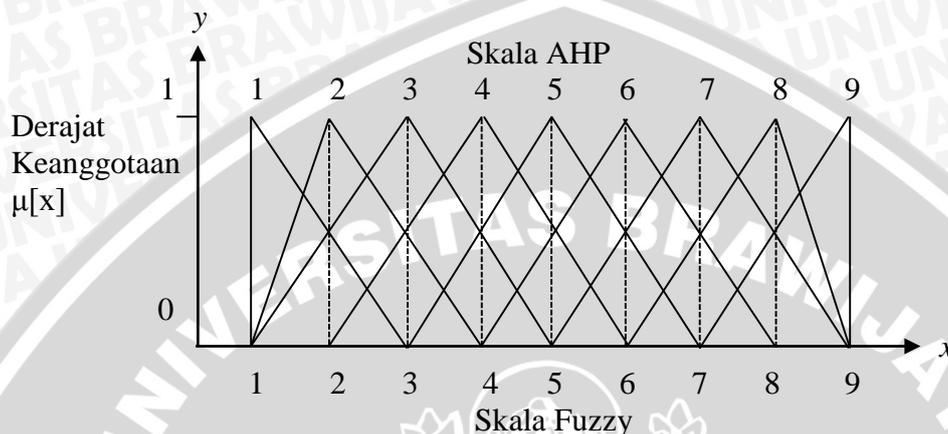
Tabel 2.7. Fuzzifikasi Perbandingan Kepentingan Antara 2 (dua) Kriteria

Skala AHP	Skala <i>Fuzzy</i>	<i>Invers</i> Skala <i>Fuzzy</i>
1	1 = (1,1,1) = jika diagonal 1 = (1,1,3) = selainnya	(1/3, 1/1, 1/1)
3	3 = (1, 3, 5)	(1/5, 1/3, 1/1)
5	5 = (3, 5, 7)	(1/7, 1/5, 1/3)
7	7 = (5, 7, 9)	(1/9, 1/7, 1/5)
9	9 = (7, 9, 9)	(1/9, 1/9, 1/7)
2	2 = (1, 2, 4)	(1/4, 1/2, 1/1)
4	4 = (2, 4, 6)	(1/6, 1/4, 1/2)
6	6 = (4, 6, 8)	(1/8, 1/6, 1/4)
8	8 = (6, 8, 9)	(1/9, 1/8, 1/6)

Sumber: [ANS-12]



Pada skala *Fuzzy* memiliki tiga variabel sebagai nilai bandingnya yaitu l untuk *lower bound*, m untuk *middle*, dan u untuk *upper bound*. Setengah diagonal matriks adalah nilai *inverse* dari setengah diagonal lainnya. Sehingga nilai dari $[l, m, u]^{-1} = [1/u, 1/m, 1/l]$. Skala fuzzifikasi perbandingan kepentingan antara 2 (dua) kriteria pada Tabel 2.5 digambarkan dalam bentuk grafik seperti pada Gambar 2.9.



Gambar 2.9. Grafik Fuzzifikasi Skala AHP
Sumber: [ANS-12]

2.6.5. Langkah Kerja FAHP

Adapun cara kerja F-AHP adalah [JAS-11]:

1. Membuat struktur hirarki masalah yang akan diselesaikan dan menentukan perbandingan matriks berpasangan antar kriteria skala TFN.
2. Menentukan nilai sintesis *fuzzy* (S_i) prioritas dengan rumus

$$S_i = \sum_{j=1}^g M_i^j \times \frac{1}{\sum_{i=1}^h \sum_{j=1}^g M_i^j} \dots \dots \dots (2-24)$$

dimana:

$$\sum_{j=1}^g M_i^j = \sum_{j=1}^g l_j, \sum_{j=1}^g m_j, \sum_{j=1}^g u_j \dots \dots \dots (2-25)$$

Sedangkan

$$\frac{1}{\sum_{i=1}^h \sum_{j=1}^g M_i^j} = \frac{1}{\sum_{i=1}^h u_i, \sum_{i=1}^h m_i, \sum_{i=1}^h l_i} \dots \dots \dots (2-26)$$

dimana:

- M = objek (kriteria, subkriteria, atau alternatif),
- i = baris ke-i,



- j = kolom ke-j,
- l = nilai *lower*,
- m = nilai *middle*,
- u = nilai *upper*
- g = h = banyaknya objek (kriteria, subkriteria, atau alternatif)

3. Menentukan nilai vektor (V) dan nilai ordinat defuzzifikasi (d'). jika hasil yang diperoleh pada setiap matrik fuzzy, $M_2 \geq M_1$ ($M_2 = (l_2, m_2, u_2)$ dan $M_1 = (l_1, m_1, u_1)$) maka nilai dari vector dapat dirumuskan sebagai berikut:

$V(M_2 \geq M_1) = \sup[\min^*(\mu_{M_1}(x), \min(\mu_{M_2}(y)))]$ atau sama dengan grafik pada persamaan (2-7) berikut:

$$V(M_2 \geq M_1) = \begin{cases} 1 & \text{if } m_2 \geq m_1, \\ 0 & \text{if } l_1 \geq u_2, \\ \frac{l_1 - u_2}{(m_2 - u_2) - (m_1 - l_1)}, & \text{lainnya} \end{cases} \dots\dots (2-27)$$

Jika hasil nilai fuzzy lebih besar dari k, M_i , ($i = 1, 2, \dots, k$) maka nilai vektor dapat didefinisikan sebagai berikut:

$$V(M \geq M_1, M_2, \dots, M_k) = V(M \geq M_1) \text{ dan } V(M \geq M_2) \text{ dan } V(M \geq M_k) = \min V(M \geq M_i) \dots\dots\dots (2-28)$$

asumsikan bahwa

$$d'(A_i) = \min V(S_i \geq S_k) \dots\dots\dots (2-29)$$

untuk $k = 1, 2, \dots, n$; $k \neq i$, maka diperoleh nilai bobot vektor

$$W' = (d'(A_1), d'(A_2), \dots, d'(A_n))^T \dots\dots\dots (2-30)$$

Dimana $A_i = 1, 2, \dots, n$ adalah elemen keputusan.

4. Normalisasi nilai bobot vektor fuzzy (W). Setelah dilakukan normalisasi dari persamaan (2-28) maka nilai bobot vektor yang ternormalisasi adalah seperti rumus berikut:

$$W (d(A_1), d(A_2), \dots, d(A_n))^T \dots\dots\dots (2-31)$$

dimana W adalah bilangan non fuzzy.

2.7. Diagram Alir Data (Data Flow Diagram)

Data Flow Diagram merupakan gambaran suatu sistem yang telah ada atau sistem baru yang akan dikembangkan secara logika tanpa mempertimbangkan lingkungan fisik dimana data tersebut mengalir dan akan disimpan [HAR-05]. Oleh

karena dapat menggambarkan arus data di dalam sistem dengan terstruktur dan jelas maka DFD juga merupakan dokumentasi dari sistem yang baik. Sifat-sifat dari diagram aliran data adalah [HAR-05]:

- a. Grafik (dapat dipecah-pecah).
- b. Berdimensi banyak.
- c. Menunjukkan aliran data dan bukan aliran *control*.

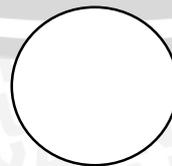
Diagram aliran data dapat dibagi menjadi beberapa tingkat yaitu; diagram konteks, diagram nol, dan diagram rinci. Berikut penjelasan masing-masing diagram:

- a. Diagram konteks, yaitu diagram tingkat tertinggi yang menyajikan sistem secara keseluruhan. Dalam diagram konteks hanya berisi proses dan eksternal entiti saja, tidak terdapat *data store*.
- b. Diagram nol, yaitu diagram yang lebih luas dari pada diagram konteks. Kegiatan-kegiatan utamanya digambarkan dalam diagram ini termasuk *data store* yang digunakan.
- c. Diagram rinci, yaitu diagram aliran data yang menjelaskan secara terperinci setiap proses yang ada pada diagram nol. Diagram ini dapat dibagi lagi menjadi diagram rinci yang lebih kecil apabila diperlukan.

Ada 4 jenis symbol yang digunakan dalam pembuatan diagram aliran data, yaitu:

- a. Lingkaran Proses (*Process*)

Menggambarkan suatu sistem atau proses aliran yang menunjukkan apa yang dikerjakan sistem yaitu mentransformasikan aliran data masuk menjadi aliran data keluar serta mengubah atau mengolah data. Setiap proses memiliki nama khusus dan nomor tersebut digambarkan di dalam lingkaran yang menyatakan proses tersebut. Notasi *Process* dapat ditunjukkan pada Gambar 2.10.



Gambar 2.10. *Process*
Sumber:[HAR-05]

b. Entitas Luar (*External Entity*)

Menggambarkan entitas luar yang berhubungan dengan sistem yang akan memberikan suatu masukan atau menerima keluaran dari sistem. Notasi *External Entity* dapat ditunjukkan pada Gambar 2.11.



Gambar 2.11. *External Entity*
Sumber: [HAR-05]

c. Aliran Data (*Data Flow*)

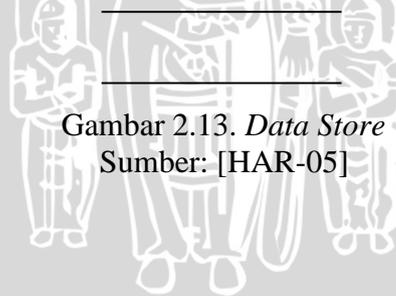
Menunjukkan aliran data yang masuk atau keluar dari sistem atau data yang mengalir di antara proses *data store* dan entitas. Notasi *Data Flow* dapat ditunjukkan pada Gambar 2.12.



Gambar 2.12. *Data Flow*
Sumber: [HAR-05]

d. Penyimpanan Data (*Data Store*)

Merupakan tempat penyimpanan data atau mengambil data yang akan diproses lebih lanjut. Notasi *Data Store* dapat ditunjukkan pada Gambar 2.13.



Gambar 2.13. *Data Store*
Sumber: [HAR-05]