

BAB II

KAJIAN PUSTAKA

2.1 Teori Dasar

2.1.1 Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan buatan adalah salah satu cabang ilmu pengetahuan berhubungan dengan pemanfaatan mesin untuk memecahkan persoalan yang rumit dengan cara yang lebih manusiawi. Hal ini biasanya dilakukan dengan mengikuti/mencontoh karakteristik dan analogi berpikir dari kecerdasan/inteligensia manusia, dan menerapkannya sebagai algoritma yang dikenal oleh komputer. Dengan suatu pendekatan yang kurang lebih fleksibel dan efisien dapat diambil tergantung dari keperluan, yang mempengaruhi bagaimana wujud dari perilaku kecerdasan buatan. AI biasanya dihubungkan dengan ilmu komputer, akan tetapi juga terkait erat dengan bidang lain seperti matematika, psikologi, pengamatan, biologi, filosofi, dan yang lainnya. Kemampuan untuk mengkombinasikan pengetahuan dari semua bidang ini pada akhirnya akan bermanfaat bagi kemajuan dalam upaya menciptakan suatu kecerdasan buatan.

Pengertian lain dari kecerdasan buatan adalah bagian ilmu komputer yang membuat agar mesin komputer dapat melakukan pekerjaan seperti dan sebaik yang dilakukan manusia (T.Sutojo, Edy Mulyanto, 2010:1). Pada generasi awal

diciptakannya, komputer hanya berfungsi sebagai alat hitung saja. Namun seiring dengan perkembangan jaman, maka peran komputer semakin mendominasi kehidupan manusia. Komputer tidak lagi hanya digunakan sebagai alat hitung, lebih dari itu, komputer diharapkan untuk dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan oleh manusia.

Manusia bisa menjadi pandai dalam menyelesaikan segala permasalahan di dunia ini karena manusia mempunyai pengetahuan dan pengalaman. Pengetahuan diperoleh dari belajar. Semakin banyak bekal pengetahuan yang dimiliki oleh seseorang tentu saja diharapkan akan lebih mampu dalam menyelesaikan permasalahan. Namun bekal pengetahuan saja tidak cukup, manusia juga diberi akal untuk melakukan penalaran, mengambil kesimpulan berdasarkan pengetahuan dan pengalaman yang mereka miliki. Tanpa memiliki kemampuan untuk menalar dengan baik, manusia dengan segudang pengalaman dan pengetahuan tidak akan dapat menyelesaikan masalah dengan baik. Demikian pula dengan kemampuan menalar yang sangat baik, namun tanpa bekal pengetahuan dan pengalaman yang memadai, manusia juga tidak akan bisa menyelesaikan masalah dengan baik. Agar komputer bisa bertindak seperti dan sebaik manusia, maka komputer juga harus diberi bekal pengetahuan dan mempunyai kemampuan untuk menalar. Untuk itu AI akan mencoba untuk memberikan beberapa metoda untuk membekali komputer dengan kedua komponen tersebut agar komputer bisa menjadi mesin pintar (T.Sutojo, Edy Mulyanto, 2010: 3).

Kecerdasan Buatan (*Artificial Intelligence*) untuk pertamakali dikemukakan pada tahun 1956, di konferensi Dartmouth. Sejak saat itu, AI atau *Artificial Intelligence* terus menerus dikembangkan sebab berbagai penelitian mengenai teori-teori dan prinsip-prinsipnya juga terus berkembang.

Di era milenial seperti sekarang perkembangan *hardware* dan *software* semakin cepat dan sudah menjadi kebutuhan yang tidak terhindarkan. Berbagai produk produk *Artificial Intelligence* (AI) telah dibuat dan digunakan dalam kehidupan sehari-hari. Dengan kemajuan teknologi *hardware* yang bentuknya semakin kecil dan kemampuannya semakin tinggi serta didukung teknologi *software* yang banyak jenis dan tangguh, maka produk-produk berbasis *Artificial Intelligence* (AI) semakin melekat dengan kehidupan manusia.

Pada masa yang akan datang, *Artificial Intelligence* (AI) ditantang untuk dapat membuat suatu produk kecerdasan buatan yang hampir menyamai kecerdasan manusia. Ray Kurzweil memperkirakan bahwa hal itu akan terwujud melalui tahapan-tahapan prediksi yang dibuatnya secara bertahap hingga tahun 2099.

2.1.2 Jaringan Syaraf Tiruan

Menurut (T.Sutojo, Edy Mulyanto, 2010: 283) Jaringan syaraf tiruan adalah paradigma pengolahan informasi yang terinspirasi oleh sistem syaraf biologis, seperti proses informasi pada otak manusia. Ada beberapa beberapa tipe jaringan syaraf, namun demikian, hampir semuanya memiliki komponen-komponen yang

sama. Seperti halnya otak manusia. Jaringan syaraf juga terdiri dari beberapa *neuron*, yang bekerja serentak untuk menyelesaikan masalah tertentu, misalnya pengenalan pola atau klasifikasi data melalui proses pembelajaran dan ada hubungan antara *neuron-neuron* tersebut.

Ada beberapa kelebihan yang dimiliki oleh jaringan saraf tiruan diantaranya:

1. Belajar adaptif, yaitu kemampuan untuk mempelajari bagaimana melakukan pekerjaan berdasarkan data yang diberikan untuk pelatihan atau pengalaman awal.
2. *Self-Organization*, yaitu kemampuan membuat organisasi sendiri atau representasi dari informasi yang diterimanya selama waktu belajar.
3. *Real Time Operation*, yaitu perhitungan jaringan saraf tiruan yang dapat dilakukan secara paralel sehingga perangkat keras yang dirancang dan diproduksi secara khusus dapat mengambil keuntungan dari kemampuan ini.

Selain mempunyai beberapa kelebihan, jaringan saraf tiruan juga mempunyai beberapa kelemahan-kelemahan diantaranya:

1. Tidak efektif jika digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi.
2. Tidak efisien jika digunakan untuk melakukan operasi algoritma aritmatika, operasi logika, dan simbolis.
3. Membutuhkan pelatihan untuk dapat beroperasi sehingga bila jumlah datanya besar, waktu yang digunakan untuk proses pelatihan sangat lama.

Salah satu elemen yang menentukan baik tidaknya suatu mode jaringan saraf tiruan adalah hubungan antar-*neuron* atau arsitektur jaringan. *Neuron-neuron* tersebut terkumpul dalam lapisan-lapisan yang disebut *neuron layers*. Terdapat 3 bagian lapisan penyusun jaringan saraf tiruan, (T.Sutojo, Edy Mulyanto, 2010: 292) yaitu:

1. Lapisan *Input* (*Input Layer*)

Unit-unit dalam lapisan ini disebut unit-unit *input* yang bertugas menerima pola *input*-an dari luar yang menggambarkan suatu permasalahan.

- 2 Lapisan Tersembunyi (*Hidden Layer*)

Unit-unit dalam lapisan ini disebut unit-unit tersembunyi, yang mana nilai *output*-nya tidak dapat diamati secara langsung.

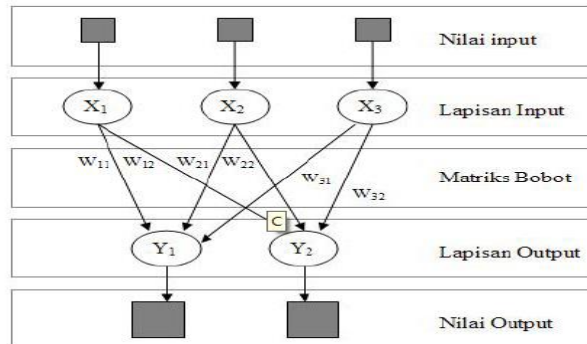
3. Lapisan *Output* (*Output Layer*)

Unit-unit dalam lapisan ini disebut unit-unit *output*, yang merupakan solusi jaringan saraf tiruan terhadap suatu permasalahan.

Beberapa arsitektur jaringan yang sering digunakan dalam jaringan saraf tiruan antara lain :

1. Jaringan Lapisan Tunggal

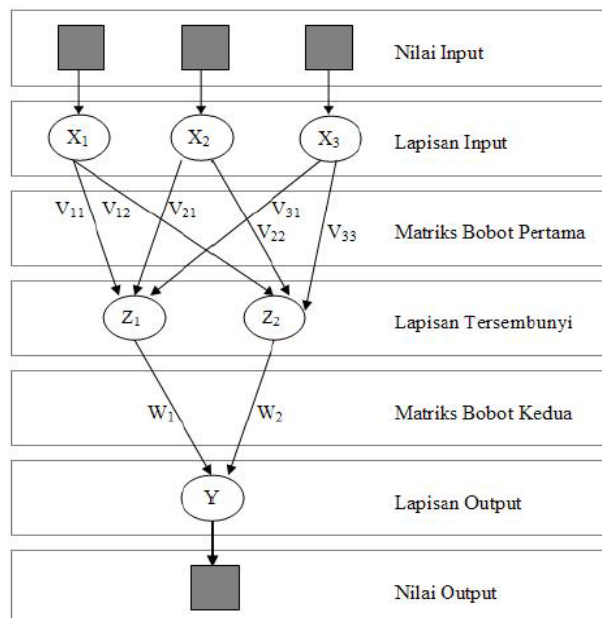
Jaringan ini terdiri dari 1 lapisan *input* dan 1 lapisan *output*, yang mana setiap unit dalam lapisan *input* selalu terhubung dengan setiap unit yang terdapat pada lapisan *output*. Jaringan ini menerima *input* kemudian mengolahnya menjadi *output* tanpa melewati lapisan tersembunyi. Contoh jaringan saraf tiruan yang menggunakan jaringan ini adalah *ADALINE*, *Hopfield*, dan *Perceptron*.



Gambar 2. 1 Jaringan Lapisan Tunggal
Sumber: (T.Sutojo, Edy Mulyanto, 2010:293)

2. Jaringan Lapisan Banyak

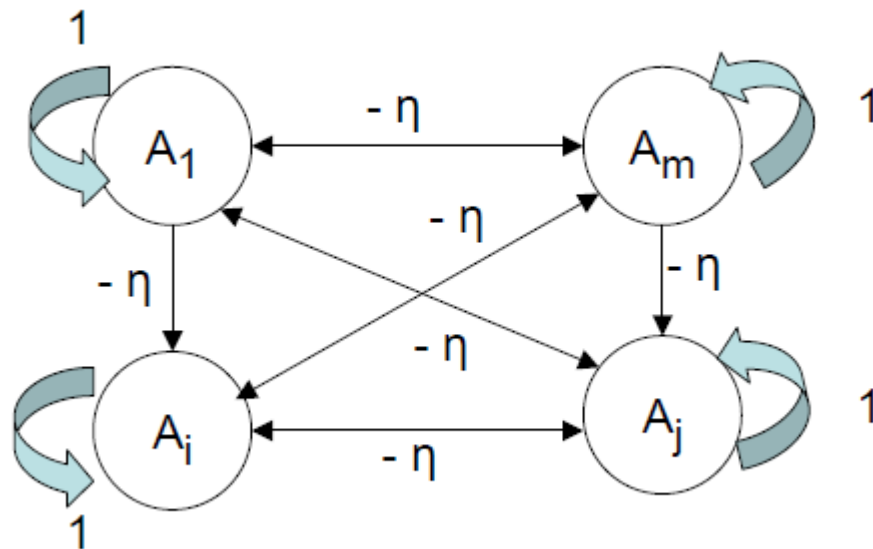
Jaringan ini mempunyai 3 jenis lapisan, yaitu lapisan *input*, lapisan tersembunyi, dan lapisan *output*. Jaringan ini dapat menyelesaikan permasalahan yang lebih kompleks dibandingkan dengan jaringan lapisan tunggal. Contoh jaringan saraf tiruan yang menggunakan jaringan ini adalah *MADALINE*, *backpropagation*, dan *Neocognitron*.



Gambar 2. 2 Jaringan Lapisan Banyak
Sumber: (T.Sutojo, Edy Mulyanto, 2010:294)

3. Jaringan dengan Lapisan Kompetitif

Jaringan ini memiliki bobot yang telah ditentukan dan tidak memiliki proses pelatihan. Jaringan ini digunakan untuk mengetahui *neuron* pemenang dari sejumlah *neuron* yang ada sehingga sekumpulan *neuron* bersaing untuk mendapatkan hak menjadi aktif. Contoh jaringan saraf tiruan yang menggunakan jaringan ini adalah *Learning Vector Quantization (LVQ)*.



Gambar 2. 3 Jaringan Lapisan Kompetitif
Sumber: (T.Sutojo, Edy Mulyanto, 2010:295)

2.1.3 Sistem Pakar (*Expert System*)

Sistem pakar mulai dikembangkan pada pertengahan 1960, ditandai dengan lahirnya sistem pakar pertama bernama *General-purpose Problem Solver (GPS)* yang dikembangkan oleh Newel dan Simon. Kemudian bermunculan sistem pakar lain di berbagai bidang seperti *MYCIN* untuk diagnosis penyakit, *DENDRAL*

untuk mengidentifikasi struktur molekul campuran yang tak dikenal, *XCON* & *XSEL* untuk membantu konfigurasi sistem komputer besar, *SOPHIE* untuk analisis sirkuit elektronik, *Prospector* digunakan di bidang geologi untuk membantu mencari dan menemukan deposit, *FOLIO* digunakan untuk membantu memberikan keputusan bagi seorang manajer dalam masalah stok dan investasi, *DELTA* dipakai untuk pemeliharaan lokomotif listrik diesel, dan sebagainya. (T.Sutojo, Edy Mulyanto, 2010: 159).

Sistem pakar adalah sebuah sistem yang menggunakan pengetahuan manusia di mana pengetahuan tersebut dimasukkan ke dalam sebuah komputer dan kemudian digunakan untuk menyelesaikan masalah-masalah yang biasanya membutuhkan kepakaran atau keahlian manusia. Dengan bantuan sistem pakar, seseorang yang bukan pakar dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar (Ernawati, 2017).

Sistem pakar mencari dan memanfaatkan informasi yang relevan dari pengguna dan dari basis pengetahuan yang tersedia untuk membuat rekomendasi. Sistem pakar memberikan nilai tambah pada teknologi untuk membantu dalam menangani era informasi yang semakin canggih (Deby Saputra, Uning Lestari, 2016).

Menurut (Soepomo, 2014) secara umum, sistem pakar (*expert system*) adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer, agar komputer dapat menyelesaikan masalah seperti yang biasa dilakukan oleh para ahli. Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu

permasalahn tertentu dengan meniru kerja dari para ahli. Dengan sistem pakar ini, orang awampun dapat menyelesaikan masalah yang cukup rumit yang sebenarnya hanya dapat diselesaikan dengan bantuan para ahli. Bagi para alhi, sistem pakar juga akan membantu aktivitasnya sebagai asisten yang sangat berpengalaman.

Suatu sistem dikatakan sebagai sistem pakar jika memiliki ciri-ciri sebagai berikut (T.Sutojo, Edy Mulyanto, 2010: 162):

1. Terbatas pada domain keahlian tertentu
2. Dapat memberikan penalaran untuk data-data yang tidak lengkap atau tidak pasti
3. Dapat menjelaskan alasan-alasan dengan cara yang dapat dipahami
4. Bekerja berdasarkan kaidah tertentu
5. Mudah dimodifikasi
6. Basis pengetahuan dan mekanisme inferensi diletakkan terpisah
7. Keluarannya (*output*) bersifat anjuran
8. Sistem dapat mengaktifkan kaidah secara terpisah secara searah, sesuai dengan dialog dengan pengguna

Representasi pengetahuan merupakan metode yang digunakan untuk mengodekan pengetahuan dalam sebuah sistem pakar yang berbasis pengetahuan. Hal ini dimaksudkan untuk menangkap sifat-sifat penting dari suatu masalah sehingga informasi itu dapat diakses oleh prosedur pemecahan masalah. Bahasa representasi harus dirancang agar fakta-fakta dan pengetahuan lain yang terkandung di dalamnya dapat digunakan untuk penalaran.

Tujuan dari sistem pakar adalah memindahkan kepakaran dari seorang pakar ke dalam komputer, kemudian ditransfer kepada orang lain yang bukan pakar.

Proses ini melibatkan empat kegiatan, yaitu:

1. Akuisisi pengetahuan (dari pakar atau sumber lain)
2. Representasi pengetahuan pada komputer
3. Inferensi pengetahuan
4. Pemindahan pengetahuan ke pengguna

2.1.4 Logika Fuzzy

Fuzzy Logic adalah salah satu komponen pembentuk *soft computing*. *Fuzzy Logic* pertama kali diperkenalkan oleh Prof. Lotfi Zadeh pada tahun 1965. Dasar *Fuzzy Logic* adalah teori himpunan *Fuzzy*. Pada teori himpunan *Fuzzy*, peranan derajat keanggotaan sebagai penentu keberadaan elemen dalam suatu himpunan sangatlah penting. Nilai keanggotaan atau derajat keanggotaan atau *membership function* menjadi ciri utama dalam penalaran dengan *Fuzzy Logic* tersebut *Fuzzy Logic* dapat dianggap sebagai kotak hitam yang berhubungan antara ruang *input* menuju ruang *output*. Kotak hitam tersebut berisi cara atau metode yang dapat digunakan untuk mengolah data input menjadi output dalam bentuk informasi yang baik. (Firmansyah, 2017)

Kata-kata yang digunakan dalam *fuzzy logic* memang tidak sepresisi bilangan, namun kata-kata jauh lebih dekat dengan intuisi manusia. Dengan *fuzzy*

logic, sistem kepakaran manusia bisa di implementasikan ke dalam bahasa mesin dengan mudah dan efisien.



Gambar 2. 4 Model *Black-Box*
Sumber: (Sri Kusumadewi, 2010)

Dari sekian banyak alternatif yang tersedia, sistem *fuzzy* sering kali menjadi pilihan terbaik. Menurut Lofti A. Zadeh, dalam hampir setiap kasus, anda dapat membangun sistem yang dapat menggantikan *black-box* diatas tanpa menggunakan *fuzzy logic*. Namun demikian bila anda menggunakan *fuzzy logic*, rancang bangun sistem bisa anda lakukan lebih cepat dan efisien.

Fuzzy logic memiliki banyak kelebihan, yaitu dapat mengontrol sistem yang kompleks, non-linier dan sistem yang sulit di representasikan secara matematis. Beberapa alasan mengapa orang menggunakan *fuzzy logic*, antara lain:

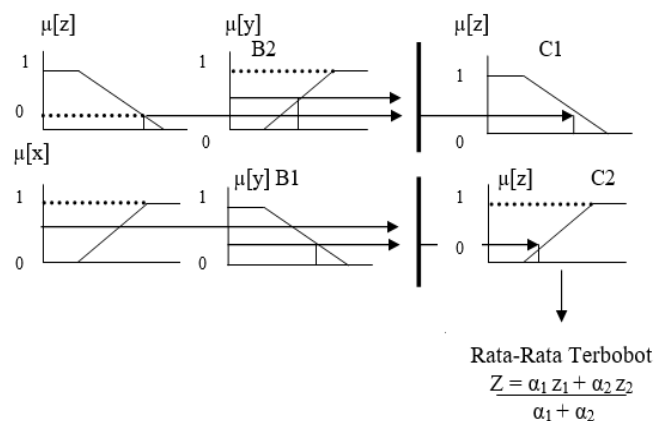
1. Konsep *fuzzy logic* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. *Fuzzy logic* sangat fleksibel.
3. *Fuzzy logic* memiliki toleransi terhadap data-data yang tidak tepat.
4. *Fuzzy logic* mampu memodelkan fungsi-fungsi non-linear yang sangat kompleks.
5. *Fuzzy logic* dapat membangun dan mengaplikasikan pengalaman-pengalaman para pakar secara langsung tanpa harus melalui pelatihan.

6. *Fuzzy logic* dapat bekerja sama dengan teknik-teknik kendali secara konvensional.
7. *Fuzzy logic* didasarkan pada bahasa alami.

Fuzzy Logic sendiri memiliki beberapa metode dalam penyelesaiannya, seperti metode Tsukamoto, Mamdani dan Sugeno. Berikut ini masing-masing penjelasannya:

1. Metode Tsukamoto

Menurut (Sri Kusumadewi, 2010:31) Metode Tsukamoto sendiri merupakan perluasan dari penalaran monoton, setiap konsekuen pada aturan yang berbentuk *IF-Then* harus direpresentasikan dengan suatu himpunan *fuzzy* dengan fungsi keanggotaan yang monoton. Sebagai hasilnya, *output* hasil inferensi dari tiap - tiap aturan diberikan secara tegas (*crisp*). Hasil akhirnya diperoleh dengan menggunakan rata - rata terbobot.



Gambar 2. 5 Inferensi dengan Menggunakan Metode Tsukomoto
Sumber: (Sri Kusumadewi, 2010:32)

2. Metode Mamdani

Menurut (Sri Kusumadewi, 2010:37) Metode Mamdani sering disebut dengan metode *Max-Min*. Metode ini diperkenalkan oleh Ebrahim Mamdani pada tahun 1975. Untuk mendapatkan *output*, diperlukan 4 tahapan :

1) Pembentukan himpunan *fuzzy*

Pada metode mamdani, baik variabel *input* atau variabel *output* di bagi menjadi satu atau lebih himpunan *fuzzy*.

2) Aplikasi fungsi implikasi

Pada metode mamdani, fungsi implikasi yang digunakan adalah *Min*.

3) Komposisi aturan

Apabila sistem terdiri dari beberapa aturan, maka inferensi di peroleh dari kumpulan dan korelasi antar aturan. Ada 3 metode yang digunakan dalam melakukan inferensi sistem *fuzzy*, yaitu: *Max*, *Additive* dan Probabilistik OR.

a. Metode *Maximum*

Solusi himpunan *fuzzy* diperoleh dengan cara mengambil nilai maksimum aturan, kemudian menggunakannya untuk memodifikasi daerah *fuzzy*, dan mengaplikasikannya ke *output* dengan menggunakan operator OR (*union*). Secara umum dapat dituliskan:

$\mu_{sf}[xi] \leftarrow \max(\mu_{sf}[xi], \mu_{kf}[xi])$ dengan:

$\mu_{sf}[xi]$ = nilai keanggotaan solusi *fuzzy* sampai aturan ke-i;

$\mu_{kf}[xi]$ = nilai keanggotaan konsekuen *fuzzy* aturan ke-i;

Misalkan ada 3 aturan (proposisi) sebagai berikut:

[R1] IF Biaya Produksi RENDAH And Permintaan NAIK

THEN Produksi Barang BERTAMBAH;

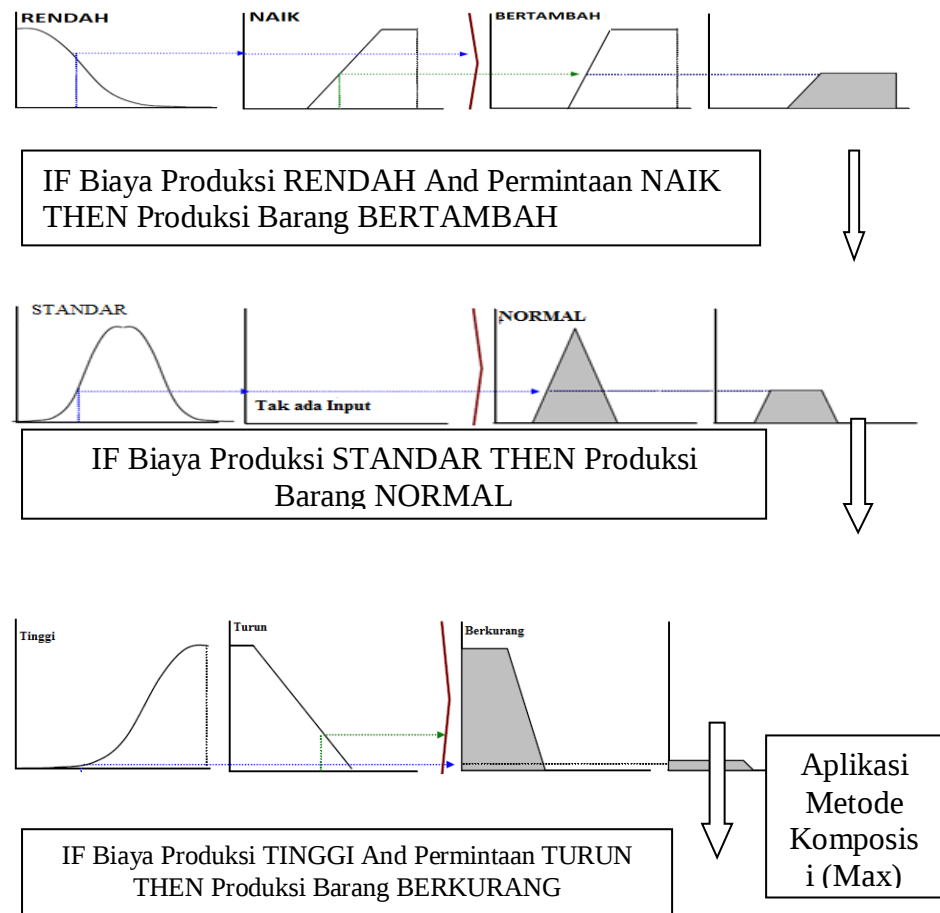
[R2] IF Biaya Produksi STANDAR

THEN Produksi Barang NORMAL;

[R3] IF Biaya Produksi TINGGI And Permintaan TURUN

THEN Produksi Barang BERKURANG;

Proses inferensi dengan menggunakan metode *Max* dalam melakukan komposisi aturan seperti terlihat pada gambar di bawah ini. Apabila digunakan fungsi implikasi *MIN*, maka metode komposisi ini sering disebut dengan nama *MAX-MIN* atau *MIN-MAX* atau MAMDANI.



Gambar 2. 6 Komposisi Aturan *Fuzzy*: Metode Max
Sumber: (Sri Kusumadewi, 2010:39)

b. Metode *Additive (sum)*

Solusi himpunan *fuzzy* di peroleh dengan cara melakukan *bounded-sum* terhadap *output* semua *output* daerah *fuzzy*.

$\mu_{sf}[x_i] \leftarrow \min(1, \mu_{sf}[x_i] + \mu_{kf}[x_i])$ dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi *fuzzy* sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuen *fuzzy* aturan ke-i;

c. Metode Probabilistik *OR*

Solusi himpunan *fuzzy* di peroleh dengan cara melakukan *product* terhadap semua *output* daerah *fuzzy*. Secara umum di tuliskan:

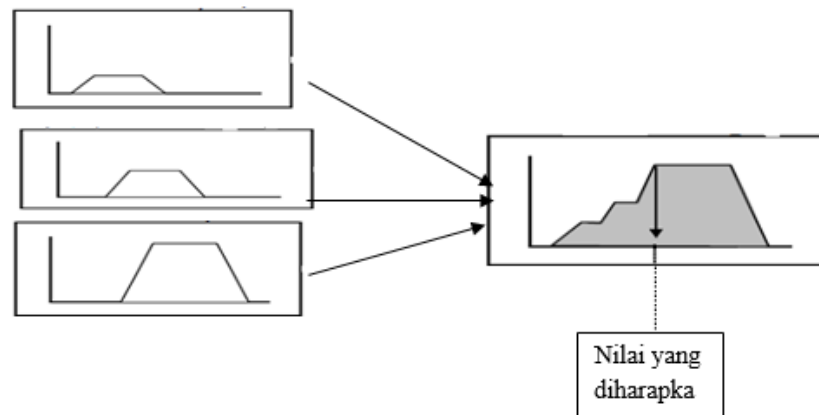
$\mu_{sf}[x_i] \leftarrow (\mu_{sf}[x_i] + \mu_{kf}[x_i]) - (\mu_{sf}[x_i] * \mu_{kf}[x_i])$ dengan:

$\mu_{sf}[x_i]$ = nilai keanggotaan solusi *fuzzy* sampai aturan ke-i;

$\mu_{kf}[x_i]$ = nilai keanggotaan konsekuen *fuzzy* aturan ke-i;

4) Penegasan (*Defuzzy*)

Input dari proses *defuzzy* adalah suatu himpunan *fuzzy* yang di peroleh dari komposisi aturan aturan *fuzzy*, sedangkan *output* yang di hasilkan merupakan suatu bilangan pada domain himpunan *fuzzy* tersebut. Sehingga jika diberikan suatu himpunan *fuzzy* dalam range tertentu, maka harus dapat diambil suatu nilai *crisp* tertentu sebagai *output*, seperti pada gambar dibawah ini.



Gambar 2. 7 Proses Defuzzifikasi
Sumber: (Sri Kusumadewi, 2010:40)

Ada beberapa metode defuzzifikasi pada komposisi aturan MAMDANI, antara lain:

a. Metode *Centroid (Composite Moment)*

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil titik pusat (z^*) daerah *fuzzy*. Secara umum dirumuskan:

$$Z^* = \frac{\int z \mu(z) dz}{\int \mu(z) dz}$$

Rumus 2. 1 Metode *Centroid* untuk variabel kontinu.

Persamaan di atas digunakan jika *output* defuzzifikasi sudah berupa nilai tegas. Apabila *output fuzzy* diskrit maka persamaan yang digunakan adalah:

$$Z^* = \frac{\sum i \mu(z_i) z_i}{\sum \mu(z_i)}$$

Rumus 2. 2 Metode *Centroid* untuk variabel diskrit

b. Metode Bisektor

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai pada domain *fuzzy* yang memiliki nilai keanggotaan setengah dari jumlah total nilai keanggotaan pada daerah *fuzzy*. Secara umum dituliskan:

	p	R_n	
Z_p sedemikian dituliskan	$\int$	$\mu(z) dz$	$= \int$
	R_l	p	

Rumus 2. 3 Bentuk umum metode Bisektor

c. Metode *Mean of Maximum (MOM)*

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai rata - rata domain yang memiliki nilai keanggotaan maksimum.

d. Metode *Largest of Maximum (LOM)*

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terbesar dari domain yang memiliki nilai keanggotaan maksimum.

e. Metode *Smallest of Maximum (SOM)*

Pada metode ini, solusi *crisp* diperoleh dengan cara mengambil nilai terkecil dari domain yang memiliki nilai keanggotaan maksimum.

3. Metode Sugeno

Menurut (Sri Kusumadewi, 2010:46) penalaran dengan metode SUGENO hampir sama dengan penalaran MAMDANI, hanya saja *output* (konsekuen) sistem tidak berupa himpunan *fuzzy*, melainkan berupa konstanta atau persamaan

linear. Metode ini diperkenalkan oleh Takagi-Sugeno Kang pada tahun 1985.

Metode ini terdiri dari 2 jenis, yaitu:

a) Model *Fuzzy* Sugeno Orde-Nol

Secara umum bentuk model *fuzzy* SUGENO Orde-Nol adalah:

IF (x1 is A1) • (x2 is A2) • (x3 is A3) • • (xN is AN) THEN $z=k$

dengan A_i adalah himpunan *fuzzy* ke-i sebagai anteseden, dan k adalah suatu konstanta (tegas) sebagai konsekuen.

b) Model *Fuzzy* Sugeno Orde-Satu

Secara umum bentuk model *fuzzy* SUGENO Orde-Satu adalah:

IF (x1 is A1) • • (xN is AN) THEN $z = p_1 \cdot x_1 + \dots + p_N \cdot x_N + q$

dengan A_i adalah himpunan *fuzzy* ke-i sebagai anteseden, dan p_i adalah suatu konstanta (tegas) ke-i dan q juga merupakan konstanta dalam konsekuen. Apabila komposisi aturan menggunakan metode SUGENO, maka defuzzifikasi dilakukan dengan cara mencari nilai rata-ratanya.

Untuk memahami logika *fuzzy* secara tepat ada beberapa hal yang harus dipahami antara lain :

1. Konsep *Fuzzy Logic*

Seringkali ditemui dalam pernyataan yang dibuat oleh seseorang, evaluasi dan suatu pengambilan keputusan. *Fuzzy system* (sistem kabur) didasari atas konsep himpunan kabur yang memetakan domain input kedalam *domain output*. Perbedaan mendasar himpunan tegas dengan himpunan kabur adalah nilai keluarannya. Himpunan tegas hanya memiliki dua nilai output yaitu nol atau satu, sedangkan himpunan kabur memiliki banyak nilai keluaran yang dikenal dengan

nilai derajat keanggotaannya. Logika *fuzzy* adalah peningkatan dari logika *Boolean* yang berhadapan dengan konsep kebenaran sebagian. Dimana logika klasik (*crisp*) menyatakan bahwa segala hal dapat diekspresikan dalam istilah *binary* (0 atau 1, hitam atau putih, ya atau tidak). Logika *fuzzy* menggantikan kebenaran *Boolean* dengan tingkat kebenaran. Logika *fuzzy* memungkinkan nilai keanggotaan antara 0 dan 1, tingkat keabuan dan juga hitam dan putih, dan dalam bentuk *linguistic*, konsep tidak pasti seperti “sedikit”, “lumayan”, dan “sangat”.

Tidak seperti logika boolean yang memiliki 2 nilai, *fuzzy logic* terdiri dari banyak nilai. *Fuzzy logic* menangani derajat keanggotaan dan derajat kebenaran. *Fuzzy logic* menggunakan nilai berkelanjutan antara 0 (sepenuhnya salah) dan 1 (sepenuhnya benar). Tidak hanya hitam dan putih, *fuzzy logic* mencakup spektrum warna, menandakan bahwa elemen-elemen bisa sebagian benar dan sebagian salah pada waktu yang sama.

2. Himpunan *Fuzzy*

Menurut (Sri Kusumadewi, 2010:3) Pada himpunan tegas (*Crisp*), nilai keanggotaan suatu *item* x dalam suatu himpunan A , yang sering di tulis dengan $\mu_A(x)$, memiliki dua kemungkinan yaitu:

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

Pada himpunan *crisp*, nilai keanggotaan hanya ada dua kemungkinan, yaitu 0 dan 1, pada himpunan *fuzzy* nilai keanggotaan terletak pada rentang 0 sampai 1,

apa bila X memiliki nilai keanggotaan *fuzzy* $\mu_A(x)=0$ berarti x tidak menjadi anggota himpunan A , demikian apabila x memiliki nilai keanggotaan *fuzzy* $\mu_A(x)=1$ berarti x menjadi anggota penuh pada himpunan A . Keanggotaan *fuzzy* memberikan suatu ukuran terhadap pendapat atau keputusan, sedangkan probabilitas mengindikasikan proposisi terhadap keseringan suatu hasil bernilai benar dalam jangka panjang. Himpunan *fuzzy* memiliki dua atribut yaitu:

1. Linguistik, yaitu penamaan suatu grup yang mewakili suatu keadaan atau kondisi tertentu dengan menggunakan bahasa alami, seperti: MUDA, PAROBAYA dan TUA.
2. Numeris, yaitu suatu nilai (angka) yang menunjukkan ukuran dari suatu variabel, seperti: 40, 25, 50, dsb.

Ada beberapa hal atau istilah dasar yang perlu di ketahui dalam memahami sistem *fuzzy*, yaitu:

- a. Variabel *Fuzzy*

Variabel *fuzzy* merupakan variabel yang hendak di bahas dalam suatu sistem *fuzzy*. Contoh: umur, temperatur, permintaan dsb.

- b. Himpunan *fuzzy*

Himpunan *fuzzy* merupakan suatu grup yang mewakili suatu kondisi atau keadaan tertentu dalam suatu variabel *fuzzy*.

- c. Semesta pembicaraan

Semesta pembicaraan adalah keseluruhan nilai yang diperbolehkan untuk di operasikan dalam suatu variabel *fuzzy*. Sementara pembicaraan merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton

dari kiri ke kanan. Nilai semesta pembicaraan dapat berupa bilangan *positif* maupun *negatif*. Adakalanya nilai semesta pembicaraan ini tidak dibatasi batas atasnya.

d. Domain

Domain himpunan *fuzzy* adalah keseluruhan nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*.

Seperti halnya semesta pembicaraan, domain merupakan himpunan bilangan *real* yang senantiasa naik (bertambah) secara monoton dari kiri kekanan.

Nilai domain dapat berupa bilangan *positif* maupun *negatif*.

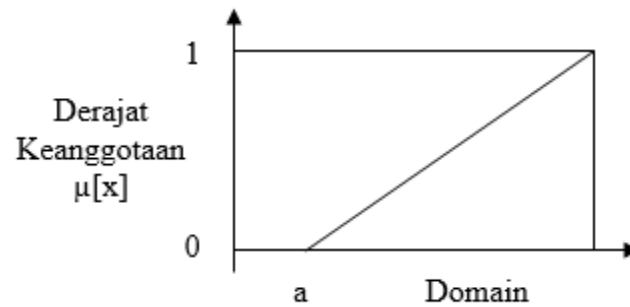
3. Fungsi Keanggotaan

Fungsi keanggotaan (*membership function*) adalah suatu kurva yang menunjukkan pemetaan titik-titik *input* data ke dalam nilai keanggotaannya (sering juga di sebut dengan derajat keanggotaan) yang memiliki interval antara 0 sampai

1. Salah satu cara yang dapat digunakan untuk mendapatkan nilai keanggotaan adalah dengan melalui pendekatan fungsi yang bisa digunakan diantaranya adalah: (Sri Kusumadewi, 2010:8)

a) Representasi Linier

Pada representasi linier, pemetaan *input* ke derajat keanggotaan digambarkan sebagai suatu garis lurus. Ada 2 keadaan himpunan *fuzzy* yang linier. Pertama, kenaikan himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi.



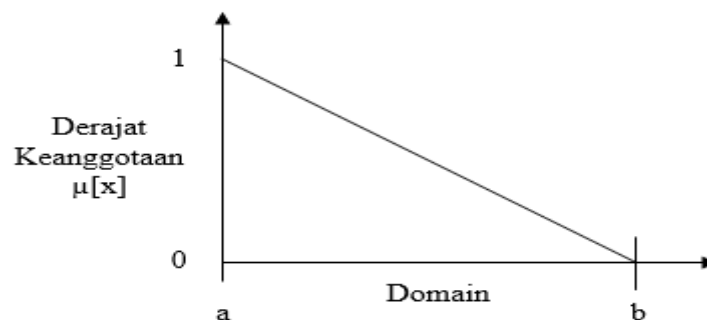
Gambar 2. 8 Representasi Linear Naik
Sumber: (Sri Kusumadewi, 2010:9)

Fungsi keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \\ (x-a) / (b-a); & a \leq x \leq b \\ 1; & x \geq b \end{cases}$$

Rumus 2. 4 Fungsi Keanggotaan
 Representasi Linier Naik

Ke dua, merupakan kebalikan yang pertama. Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan lebih rendah.



Gambar 2. 9 Representasi Linier Turun
Sumber: (Sri Kusumadewi, 2010:10)

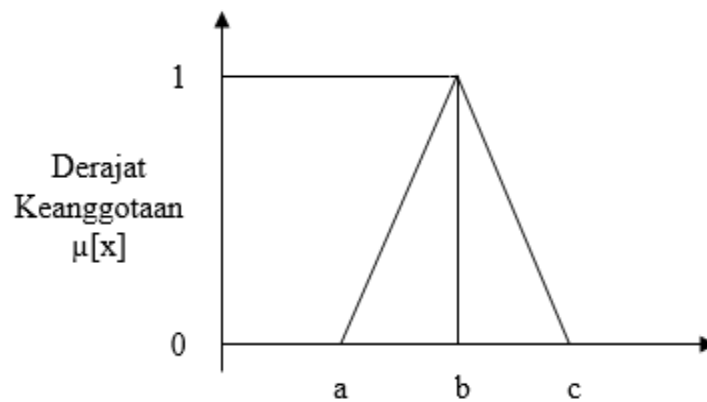
Fungsi Keanggotaan:

$$\mu[x] = \begin{cases} (b-x) / (b-a); & a \leq x \leq b \\ 0; & x \geq b \end{cases}$$

Rumus 2. 5 Fungsi keanggotaan Representasi linier turun

b) Representasi Kurva Segitiga

Kurva segitiga pada dasarnya merupakan gabungan antara 2 garis (linier).



Gambar 2. 10 Kurva Segitiga
Sumber: (Sri Kusumadewi, 2010:11)

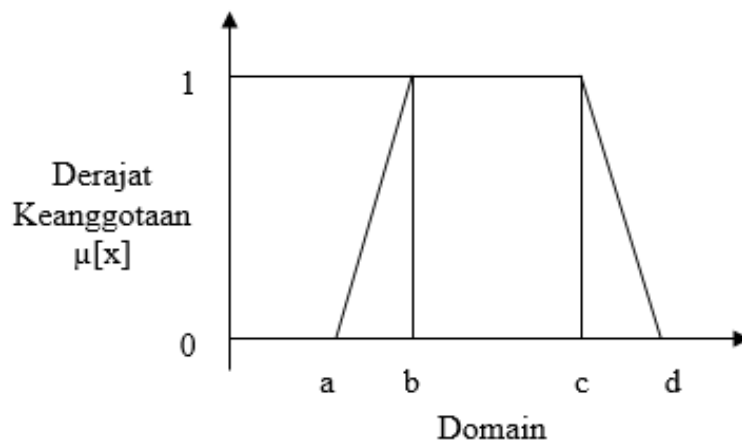
Fungsi Keanggotaan:

$$\mu[x] = \begin{cases} 0; & x \leq a \text{ atau } x \geq c \\ (x-a) / (b-a); & a \leq x \leq b \\ (b-x) / (c-b); & b \leq x \leq c \end{cases}$$

Rumus 2. 6 Fungsi keanggotaan kurva segitiga

c) Representasi Kurva Trapesium

Kurva segititiga pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1.



Gambar 2. 11 Kurva Trapesium
Sumber: (Sri Kusumadewi, 2010:13)

Fungsi Keanggotaan:

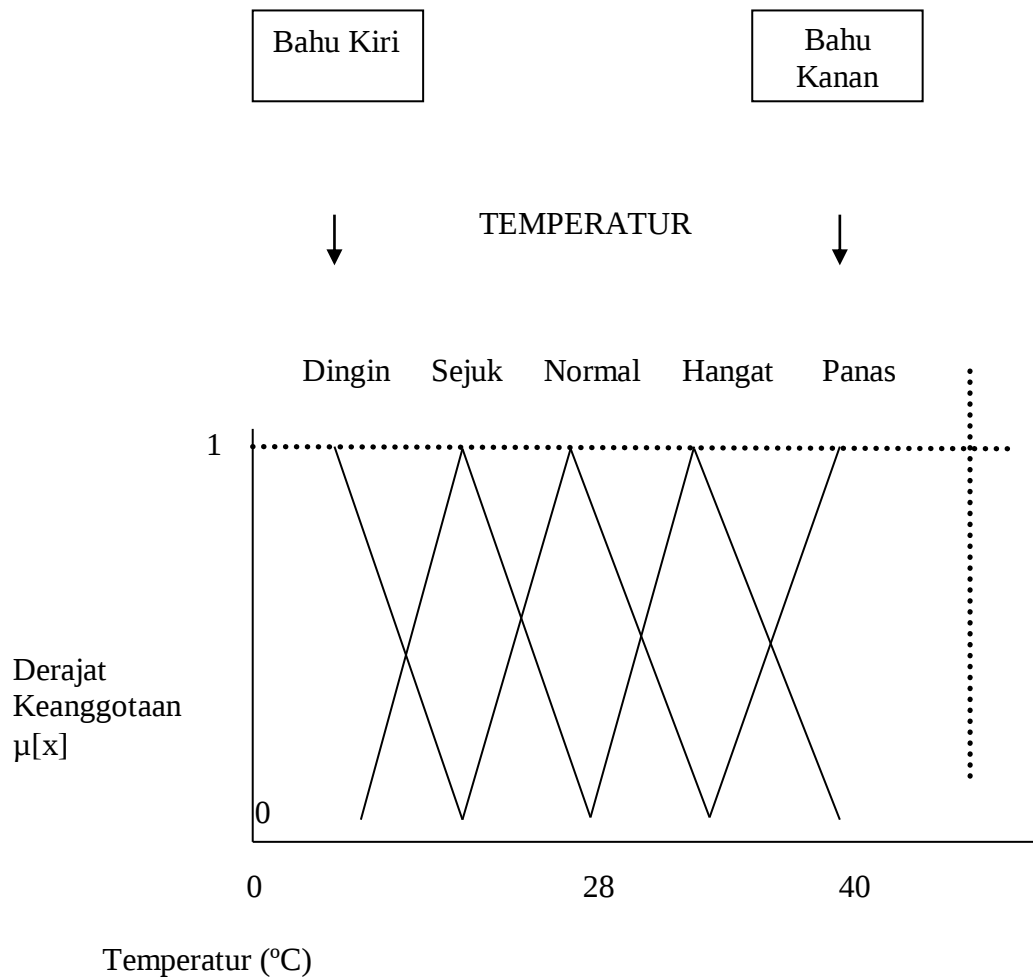
$$\mu[x] = \begin{cases} 0 ; & x \leq a \text{ atau } x \geq d \\ (x - a) / (b - a) ; & a \leq x \leq b \\ 1 ; & b \leq x \leq c \\ (d - x) / (d - c) ; & x \geq d \end{cases}$$

Rumus 2. 7 Fungsi Keanggotaan Kurva Trapesium

d) Representasi Kurva Bentuk Bahu

Daerah yang terletak di tengah tengah suatu variabel yang direpresentasikan dalam bentuk segitiga, pada sisi kanan dan kirinya akan naik dan turun. (misalkan: DINGIN bergerak ke SEJUK bergerak ke HANGAT dan bergerak ke PANAS) Tetapi terkadang salah satu sisi dari variabel tersebut tidak mengalami perubahan. Himpunan *fuzzy* “bahu”, bukan segitiga, digunakan untuk mengakhiri variabel

suatu daerah *fuzzy*. Bahu kiri bergerak dari benar ke salah, demikian juga bahu kanan bergerak dari salah ke benar.

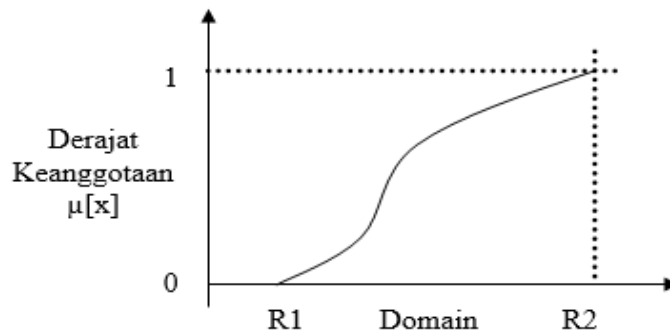


Gambar 2. 12 Representasi Kurva bentuk Bahu
Sumber: (Sri Kusumadewi, 2010:14)

e) Representasi Kurva-S

Kurva PERTUMBUHAN dan PENYUSUTAN merupakan Kurva-S atau sigmoid yang berhubungan dengan kenaikan dan penurunan permukaan secara tak linier. Kurva-S untuk PERTUMBUHAN akan bergerak dari sisi paling kiri (nilai keanggotaan = 0) ke sisi kanan (nilai keanggotaan = 1). Fungsi keanggotaannya

akan tertumpu pada 50% nilai keanggotaannya yang sering disebut dengan titik infleksi.



Gambar 2. 13 Kurva-S PERTUMBUHAN

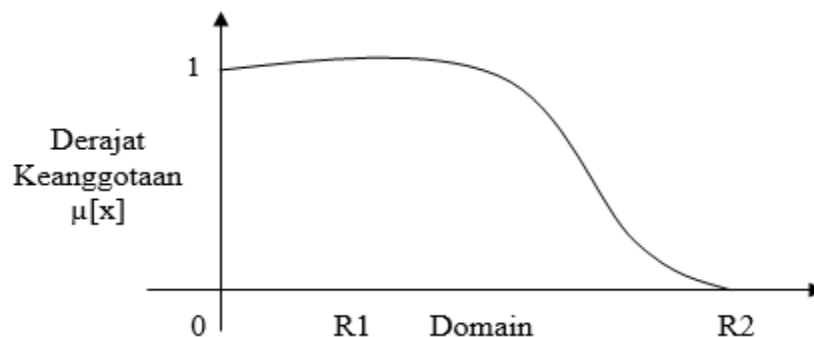
Sumber: (Sri Kusumadewi, 2010:15)

Fungsi Keanggotaan Kurva-S PERTUMBUHAN

$$S(x, \alpha, \beta, \gamma) = \begin{cases} 0 & x \leq \alpha \\ 2 \left(\frac{(x - \alpha) / (\gamma - \alpha)}{(\gamma - \alpha)} \right)^2 & \alpha \leq x \leq \beta \\ 1 - 2 \left(\frac{(\gamma - x) / (\gamma - \alpha)}{(\gamma - \alpha)} \right)^2 & \beta \leq x \leq \gamma \\ 1 & x \geq \gamma \end{cases}$$

Rumus 2. 8 Fungsi Keanggotaan Kurva-S PERTUMBUHAN

Kurva-S untuk PENYUSUTAN akan bergerak dari sisi paling kanan (nilai keanggotaan = 1) ke sisi paling kiri (nilai keanggotaan = 0).



Gambar 2. 14 Kurva-S PENURUNAN

Sumber: (Sri Kusumadewi, 2010:15)

Fungsi Keanggotaan:

$$S(x, \alpha, \beta, \gamma) = \begin{cases} 1 & x \leq \alpha \\ 1 - 2((x - \alpha) / (\gamma - \alpha))^2 & \alpha \leq x \leq \beta \\ 2((\gamma - x) / (\gamma - \alpha))^2 & \beta \leq x \leq \gamma \\ 0 & x \geq \gamma \end{cases}$$

Rumus 2. 9 Fungsi Keanggotaan Kurva-S Penurunan

4. Operator Dasar Zadeh untuk Operasi Himpunan *Fuzzy*

Menurut (Sri Kusumadewi, 2010:23) Seperti halnya himpunan konvensional, ada beberapa operasi yang didefinisikan secara khusus untuk mengkombinasi dan memodifikasi himpunan *fuzzy*. Nilai keanggotaan sebagai hasil dari operasi 2 himpunan sering di kenal dengan nama *fire strength* atau α -Predikat.

Ada 3 operator dasar yang di ciptakan oleh Zadeh, yaitu:

a) Operator *AND*

Operator ini berhubungan dengan operasi interseksi pada himpunan α -predikat sebagai hasil operasi dengan operator *AND* diperoleh dengan mengambil nilai keanggotaan terkecil antar elemen pada himpunan-himpunan yang bersangkutan.

$$\mu A \cap B = \min(\mu A[x], \mu B[y])$$

Rumus 2. 10 Bentuk umum Operator *AND*

b) Operator *OR*

Operator ini berhubungan dengan operasi *union* pada himpunan. α -predikat sebagai hasil operasi dengan operator *OR* di peroleh dengan mengambil nilai keanggotaan terbesar antar elemen pada himpunan himpunan yang bersangkutan.

$$\mu A \cap B = \max(\mu A[x], \mu B[y])$$

Rumus 2. 11 Bentuk Umum
Operator *OR*

c) Operator *NOT*

Operator ini berhubungan dengan operasi komplemen pada himpunan α -predikat sebagai hasil operasi dengan operator *NOT* di peroleh dengan mengurangi nilai keanggotaan elemen pada himpunan-himpunan yang bersangkutan dari 1.

$$\mu A' = 1 - \mu A[x]$$

Rumus 2. 12 Bentuk Umum
Operator *NOT*

5. Fungsi Implikasi

Menurut (Sri Kusumadewi, 2010:28) Tiap-tiap aturan (proposisi) pada basis pengetahuan *fuzzy* akan berhubungan dengan suatu relasi *fuzzy*. Bentuk umum dari aturan yang digunakan dalam fungsi implikasi adalah:

$$IF \ x \text{ is } A \ THEN \ y \text{ is } B$$

Dengan x dan y adalah skalar, dan A dan B adalah himpunan *fuzzy*. Proposisi yang mengikuti *IF* disebut sebagai anteseden, sedangkan yang mengikuti *THEN* disebut sebagai konsekuen. Proporsisi ini dapat di perluas dengan menggunakan operator *fuzzy*.

Secara umum ada 2 fungsi implikasi yang dapat digunakan, pertama fungsi min (minimum). Fungsi ini akan memotong *output* himpunan *fuzzy*, dan

yang kedua fungsi dot (*product*). Fungsi ini akan menskala *output* himpunan *fuzzy*.

2.2 Variabel (Indikator Masalah/Kriteria)

Variabel Penelitian adalah suatu atribut, nilai/ sifat dari objek, individu kegiatan yang mempunyai banyak variasi tertentu antara satu dan lainnya yang telah di tentukan oleh peneliti untuk di pelajari dan di cari informasinya serta di tarik kesimpulannya. (Sugiyono, 2014:38) Variabel penelitian indikator adalah sesuatu yang dapat menjadi petunjuk atau keterangan.

Variabel – variabel yang akan digunakan dalam penelitian ini terdiri dari:

a. Usia

Usia adalah variabel untuk menjadi peserta lomba pada saat mengajukan diri mendaftar seleksi.

b. Keahlian

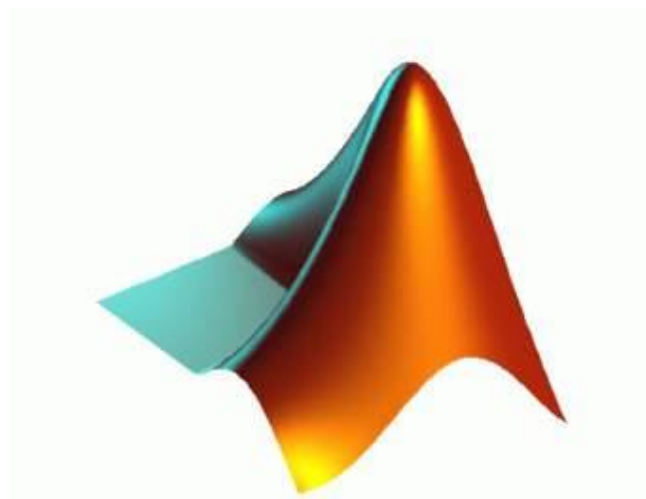
Keahlian adalah variabel berupa ketepatan peserta dalam membaca Al Qur`an dengan tajwid dan menguasai beberapa tingkatan lagu dalam tilawah.

c. Etika

Etika adalah variabel berupa tatacara peserta mengagungkan Al Qur`an dan menghayati bacaan ayat yang dibaca.

2.3 Software Pendukung

Peneliti dalam memproses data menggunakan aplikasi *Matlab* guna mendapatkan data output yang baik. *Matlab* adalah singkatan dari *MATrix LABoratory*, merupakan bahasa pemrograman yang dikembangkan oleh *The Mathwork Inc.* yang hadir dengan fungsi dan karakteristik yang berbeda dengan bahasa pemrograman lain yang sudah ada lebih dahulu seperti *Delphi*, *Basic* maupun *C++*. *Matlab* merupakan bahasa pemrograman level tinggi yang dikhususkan untuk kebutuhan komputasi teknis, visualisasi dan pemrograman seperti komputasi matematik, analisis data, pengembangan algoritma, simulasi dan pemodelan dan grafik-grafik perhitungan. Pada awalnya Matlab dibuat untuk memberikan kemudahan mengakses data matrik pada proyek *LINPACK* dan *EISPACK*. Saat ini matlab memiliki ratusan fungsi yang dapat digunakan sebagai *problem solver* baik permasalahan yang mudah maupun masalah-masalah yang kompleks dari berbagai disiplin ilmu (Tjolleng, 2017:1).



Gambar 2. 15 Logo Matlab
Sumber: (Tjolleng, 2017)

Matlab menyediakan beberapa *tool* yang dapat memudahkan pengguna, yaitu:

1. *Fuzzy Logic Toolbox*

Menurut (Ramdani, 2015) *Fuzzy logic toolbox* adalah sekumpulan *tool* yang membantu anda merancang sistem *fuzzy* untuk di aplikasikan dalam berbagai bidang, seperti *automatic control*, *signal processing*, *identification system*, *pattern recognition*, *time series prediction*, *data maining* dan bahkan *financial application*. Dengan *Fuzzy logic toolbox*, anda bisa membuat atau mengedit FIS dalam lingkungan kerja MATLAB.

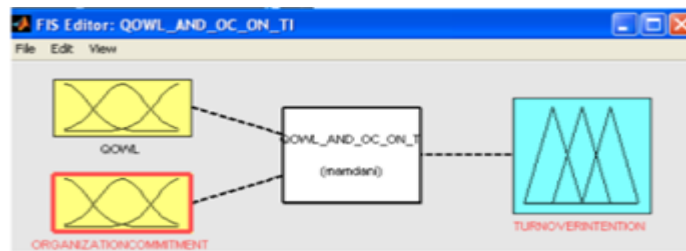
Fuzzy logic toolbox sangat *user friendly*, memungkinkan anda untuk berkreasi dengan bebas dalam rancang bangun FIS. Contohnya anda bisa mengganti fungsi –fungsi bawaan MATLAB yang di pakai dalam lima tahap pembangunan FIS dengan fungsi buatan anda sendiri, seperti Fungsi keanggotaan, operator *AND*, operator *OR*, metode implikasi, metode agregasi dan metode defuzzifikasi.

2. *Graphical User Interface*

Fuzzy logic toolbox menyediakan 5 jenis GUI untuk keperluan rancang bangun FIS:

1. *FIS editor*

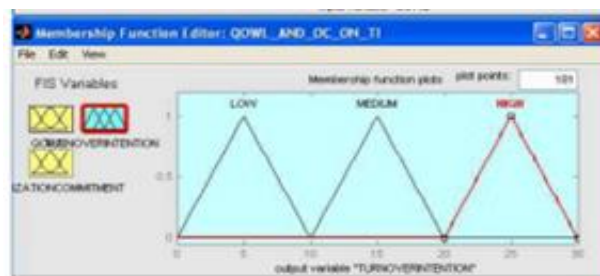
FIS Editor menampilkan informasi tingkat tinggi tentang sistem inferensi *fuzzy*. Pada MATLAB *prompt*, ketik *fuzzy* maka akan muncul *FIS editor* dengan sebuah variabel masukan dengan label *input 1* dan sebuah *output* dengan label *output 1*.



Gambar 2. 16 FIS editor Mamdani
Sumber: (Ramdani, 2015)

2. Membership function Editor

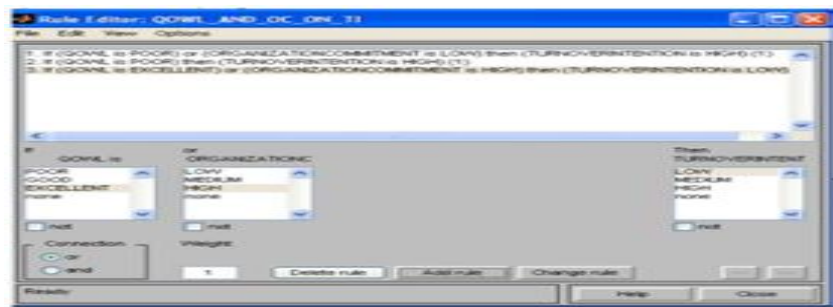
Membership Function Editor digunakan untuk mendefinisikan fungsi keanggotaan dari variabel *input* dan *output*.



Gambar 2. 17 Membership Function Editor
Sumber: (Ramdani, 2015)

3. Rule Editor

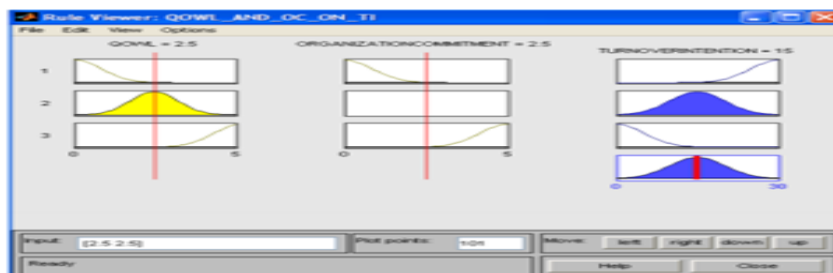
Rule Editor digunakan untuk memasukkan aturan-aturan logika yang dibuat. Berdasarkan deskripsi variabel-variabel masukan dan keluaran yang didefinisikan dalam *FIS editor*, *Rule Editor* memudahkan menyusun pernyataan-pernyataan *IF-THEN rule* secara otomatis, dengan mengklik sebuah item opsi nilai linguistic untuk tiap variabel FIS.



Gambar 2. 18 Rule Editor
Sumber: (Ramdani, 2015)

4. Rule Viewer

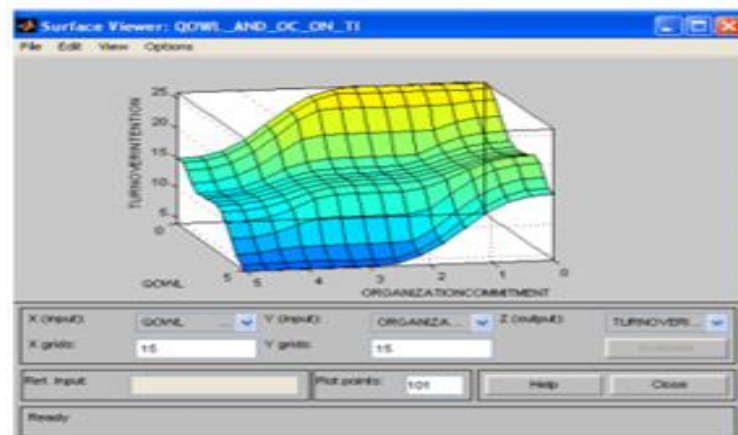
Rule viewer berfungsi untuk menampilkan proses keseluruhan dalam bentuk diagram inferensi *fuzzy*. Dari *rule viewer* dapat diketahui aturan mana yang aktif atau bagaimana suatu fungsi keanggotaan dari variabel mempengaruhi *output*.



Gambar 2. 19 Rule Viewer
Sumber: (Ramdani, 2015)

5. Surface Viewer

Surface viewer mempunyai kemampuan khusus yang sangat membantu dalam kasus dengan dua atau lebih *input* dan sebuah *output* FIS.



Gambar 2. 20 *Surface Viewer*
Sumber: (Ramdani, 2015)

Semua GUI ini saling mempengaruhi dalam arti perubahan yang dibuat dalam satu GUI akan mempengaruhi GUI yang lain. Kita bisa membuka satu persatu atau secara bersamaan seluruh GUI tersebut. Kita dapat mengubah dengan cara apapun kapasitas fungsi toolbox yang bekerja dengan menduplikasi dan mengubah nama *file* M, kemudian mengubah duplikat. Selain itu kita dapat memperkuat *toolbox* dengan memasukkan kita *M-files*. Also, *toolbox* memberikan berbagai alat interaktif yang memungkinkan kita mendapatkan pintu masuk ke sejumlah besar kapasitas melalui GUI. *Fuzzy Logic Toolbox* memungkinkan kita untuk melakukan beberapa hal, namun hal yang paling penting itu memberi kita kesempatan untuk membuat dan mengubah *system fuzzy*. Kita dapat membuat kerangka kerja ini menggunakan alat grafis atau fungsi baris perintah, atau kita dapat menghasilkan *output* segera dengan memanfaatkan baik pengelompokan atau teknik neuro adaptif. Kami dengan mudah dapat menguji kerangka *fuzzy* kita dalam lingkungan simulasi diagram blok, asalkan kita masuk ke *Simulink* (Ramdani, 2015).

2.4 Penelitian Terdahulu

Berikut ini merupakan penelitian-penelitian yang berhubungan dengan judul yang diangkat pada penelitian ini, yang digunakan untuk memperkuat dan menambah referensi penelitian.

1. **(Sherly Jayanti, 2012)** “Sistem Pendukung Keputusan Seleksi Anggota Paduan Suara Dewasa Menggunakan Metode Fuzzy Mamdani”. IJCCS, Vol.6, No.1, January 2012 ISSN: 1978-1520.

Penelitian ini membahas tentang seleksi paduan suara yang memerlukan pertimbangan dan alasan tertentu untuk menjadi anggota. Adapun hasil dari penelitian ini adalah:

- a. Membantu peningkatan jumlah skor para peserta karena range data yang dipakai cukup panjang, sehingga memungkinkan untuk meningkatkan skor peserta lebih tinggi dari hasil perhitungan.
- b. Perubahan nilai pada salah satu data akan mengubah nilai hasil perhitungan dan hasil keputusan. Sehingga tidak menutup kemungkinan seseorang akan diterima berdasarkan kriteria sebelumnya dan tidak diterima menggunakan kriteria yang baru.

2. **(Andani, 2013)** “FUZZY MAMDANI DALAM MENENTUKAN TINGKAT KEBERHASILAN DOSEN MENGAJAR”. Seminar Nasional Informatika 2013 (semnasIF 2013) ISSN: 1979-2328. Penelitian ini mengangkat masalah cara menentukan tingkat keberhasilan dosen mengajar jika hanya menggunakan dua variabel *input*, yaitu dosen dan nilai.

Penyelesaian masalah tingkat keberhasilan dosen mengajar dengan menggunakan metode *fuzzy* Mamdani yaitu menentukan variable *input* dan *output* yang merupakan himpunan tegas. Adapun hasil dari penelitian ini adalah:

- a. Diperoleh suatu model yang dapat memperlihatkan aturan keterhubungan antara motivasi dosen, persiapan mengajar dosen dan pelaksanaan perkuliahan dengan nilai mahasiswa.
- b. Logika *fuzzy* membantu dalam memberikan hasil yang tidak *crisp* dengan menggunakan konsep sifat kesamaran suatu nilai.
- c. Penelitian ini telah menunjukkan korelasi variabel dosen dengan variabel nilai, dalam menentukan tingkat keberhasilan dosen mengajar.

3. **(Firmansyah, 2017)** “FUZZY DECISION SUPPORT SYSTEM (FDSS) UNTUK SELEKSI PENERIMAAN SISWA BARU”. IJCIT (*Indonesian Journal on Computer and Information Technology*) Vol.2 No.1, Mei 2017, E-ISSN: 2549-7421. Penelitian ini membahas bagaimana metode *Fuzzy Inference System* (FIS) Mamdani dapat digunakan untuk pengambilan keputusan seleksi penerimaan siswa baru yang akan melanjutkan ke jenjang yang lebih tinggi. Adapun tujuan dari penelitian ini adalah:

- a. Untuk meningkatkan akurasi pada proses seleksi penerimaan siswa baru dibandingkan dengan proses yang dilakukan secara konvensional atau manual.

- b. Membantu memudahkan pekerjaan panitia seleksi penerimaan siswa baru pada saat melakukan seleksi, khususnya ketika ditemukan data siswa yang memiliki kesamaan jumlah nilai.

4. **(Zamroni, 2014)** “Pemanfaatan metode *Logic Mamdani* untuk SPK Penerimaan Beasiswa di SMA Muhammadiyah Sugio”. Jurnal Teknik Informatika Vol 6, No. 1, (2014) ISSN: 2085-0859. Penelitian ini membahas tentang seleksi penerimaan beasiswa khusus siswa miskin yang caranya masih manual atau menulis tangan saat mendaftar, dengan menggunakan *fuzzy Mamdani*, penelitian ini bertujuan:

- a. Memberi gambaran secara objektif dalam pemberian beasiswa di SMA Muhammadiyah 10 Sugio.
- b. Aplikasi yang dibangun dapat mempermudah pekerjaan dan meminimalisir kesalahan yang dilakukan oleh para guru dan staf administrasi di SMA Muhammadiyah 10 Sugio.

5. **(Arman, 2010)** ” Sistem Pendukung Keputusan Dengan Metode Fuzzy Logic Untuk Menseleksi Mahasiswa Penerima Beasiswa”. Jurnal Edik Informatika Penelitian Bidang Komputer Sains dan Pendidikan Informatika V2.i1 E-ISSN: 2541-3716. Penelitian ini membahas tentang seleksi penerimaan beasiswa untuk mahasiswa, dengan menggunakan metode *Fuzzy Logic Tahani*. Adapun hasil dari penelitian ini adalah:

- a. Sistem pendukung keputusan dengan Metode *Fuzzy Logic Tahani* dapat membantu pimpinan untuk mendapatkan rekomendasi dalam seleksi mahasiswa penerima beasiswa.

- b. Sistem yang dibangun dapat mempercepat proses penyeleksian beasiswa dan mampu mengurangi kesalahan dalam penentuan penerima beasiswa.
- c. Model Tahani merupakan salah satu metode yang tepat untuk digunakan pada system perkomendasian dan pendukung keputusan bagi pimpinan dalam menyelesaikan masalah yang relatif.

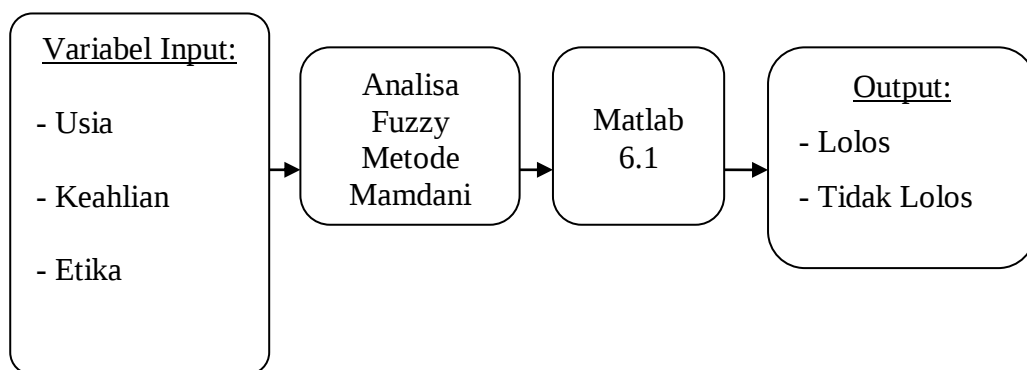
6. **(Eyupoglu, Gardashova, Allahverdiyev, & Saner, 2016)** *"The Application of Fuzzy Logic in Work Performance Satisfaction Issues"* *Procedia Computer Science* 102 (2016) 190 - 197. This study deals with job satisfaction that has been a popular research topic for decades. The interest in this topic has attracted the attention of psychologists, management scholars and, more recently, economists.

Most of the research is done in the field of job satisfaction based on statistical methods. But this method can not explain the fact that the basic aspects of job satisfaction, such as activity, independence, variety, social status, and supervision of human relationships, to name a few, are evaluated based on perceptions that do not provide appropriate numerical information. Information supported by perception can be processed more simply by using fuzzy logic. The main purpose of this work is to model job satisfaction. In this paper an evaluation of the overall job satisfaction method using fuzzy procedure.

2.5 Kerangka Pemikiran

Kerangka pemikiran yang baik akan menjelaskan secara teoritis peraturan antara variabel yang akan di teliti. Secara teoritis dapat di jelaskan antara variabel dependen dan variabel independen, bila dalam penelitian terdapat variabel intervening dan moderator maka perlu dijelaskan mengapa variabel itu ikut dilibatkan dalam penelitian. Pertautan antar variabel tersebut selanjutnya dirumuskan ke dalam bentuk paradigma penelitian. Oleh karena itu pada setiap penyusunan paradigma penelitian harus didasarkan pada kerangka berpikir (Sugiyono, 2014:60).

Dari penjelasan di atas, dengan ini penulis membuat kerangka pemikiran sebagai berikut:



Gambar 2. 21 Kerangka Pemikiran
Sumber: Data Olahan (2017)

Pada variabel input, terdapat tiga inputan yang terdiri dari usia, keahlian, dan etika, yang dijadikan tolak ukur yang nantinya akan di hitung menggunakan logika *fuzzy* metode Mamdani dan di proses menggunakan Matlab hingga akhirnya menghasilkan sebuah keputusan atau output berupa lolos dan tidak lolos.