

BAB II

TINJAUAN PUSTAKA DAN LANDASAN TEORI

2.1 Penelitian Terkait

Fuzzy Inference System (FIS) Tsukamoto telah banyak digunakan untuk melakukan penelitian dalam berbagai bidang pendidikan, bidang kesehatan, dan bidang jual beli barang.

Berikut beberapa contoh penelitian-penelitian terdahulu yang menggunakan *fuzzy inference system* tsukamoto.

a. Analisa Kelayakan Truk Pengangkut Material Alam PT. Arga Wastu Sluke – Rembang Menggunakan *Fuzzy Logic* Tsukamoto.[4]

Penelitian ini bertujuan untuk menganalisa kelayakan truk pengangkut material alam di PT. Arga Wastu Sluke – Rembang. Atribut yang diperoleh dari PT. Arga Wastu yaitu:

1. Nomor Polisi
2. Jarak yang telah ditempuh (Km)
3. Beban paling berat yang pernah diangkut (Ton)
4. Kondisi ban (%)
5. Kondisi kampas rem (%)
6. Kondisi kampas kopling (%)
7. Kondisi *peer* (%)
8. Kondisi layak atau tidak layak

Dari data diatas digunakan untuk menganalisa kelayakan truk di PT. Arga Wastu dan menghasilkan sebanyak 15 *rules* dengan operator yang digunakan adalah AND.

Berdasarkan perhitungan yang dilakukan menggunakan teknik *fuzzy logic* tsukamoto dengan himpunan fuzzy dan domain yang telah

ditentukan maka diperoleh persentase tingkat sebesar 56,66% dengan menggunakan 30 sampel data pengujian [4].

b. Sistem Pakar Untuk Diagnosa Tingkat Resiko Penyakit Rabies Pada Anjing Menggunakan Metode *Fuzzy Inference System* (FIS) Tsukamoto [5].

Penelitian ini bertujuan untuk memudahkan masyarakat yaitu pemelihara anjing, dokter hewan, petugas kesehatan untuk mendiagnosa tingkat resiko penyakit rabies berdasarkan gejala klinis yang timbul. Dari analisa data yang diperoleh oleh penulis menghasilkan 27 *rules* dengan operator yang digunakan adalah AND.

Dari hasil pengujian oleh sistem dengan jumlah kasus sebanyak 30 sampel data penelitian ini menghasilkan tingkat ketepatan diagnosa sebesar 93% [5].

c. Klasifikasi Penerimaan Beasiswa Dengan Menggunakan Logika Fuzzy Tsukamoto (Studi Kasus Politeknik Kesehatan Kementrian Kesehatan Semarang) [6].

Penelitian ini bertujuan untuk alat bantu pengambilan keputusan untuk penilaian penerimaan beasiswa pada politeknik kesehatan kementrian Semarang dengan menggunakan metode *fuzzy tsukamoto*.

Dari hasil perghitungan menggunakan metode *fuzzy tsukamoto* menghasilkan akurasi sebesar 100%. Nilai tersebut didapatkan dari jumlah peminatan yang sama dari perhitungan dengan metode *fuzzy tsukamoto* disbanding dengan jumlah sampel yang digunakan sebanyak 100 data. Sehingga aplikasi pendukung keputusan dapat digunakan pihak manajemen untuk menentukan penerimaan beasiswa pada politeknik kesehatan kementrian kesehatan Semarang agar penerimaan beasiswa yang mendapatkan dan tidak lebih optimal.

d. Implementasi Logika Fuzzy Tsukamoto Dalam Menentukan Harga Mobil Toyota Avanza 1.3 G M/T Bekas [7].

Penelitian ini bertujuan untuk menerapkan metode tsukamoto dalam menentukan harga mobil bekas avanza 1,3 G M/T berdasarkan warna, tahun pembuatan, kondisi mobil, harga beli baru dan kisaran harga jual mobil bekas. Dari atribut variabel tersebut menghasilkan 30 *rules* dengan *operator* yang digunakan adalah AND.

Dari hasil penelitian diatas dilakukan pengujian dengan menggunakan metode *MAPE* (*Meant Absolute Percentage Error*) diketahui hasil dari pengujian tersebut memiliki tingkat kesalahan sebesar 0,314%. Dengan ini calon pembeli dapat menggunakan sistem ini untuk membantu menentukan harga Toyota Avanza 1,3 G M/T bekas dipasaran.

Selain penelitian yang menggunakan metode fuzzy tsukamoto, berikut adalah beberapa contoh penelitian terdahulu yang berhubungan dengan penyakit asma.

a. Sistem Diagnosa Level Asma Menggunakan *Fuzzy Inference System*[8]

Penelitian ini bertujuan untuk membuat sebuah aplikasi yang bisa digunakan untuk mendiagnosis level penyakit asma. Metode yang digunakan dalam penelitian ini adalah *fuzzy* Mamdani. Dari hasil penelitian dilakukan pengujian di Rumah Sakit Pertamina Cilacap dengan hasil tingkat akurasi sebesar 90% dari 20 data pengujian, 18 pasien asma diagnosa sistem sama dengan diagnosa dokter.

b. Rancang Bangun Aplikasi Sistem Pakar Untuk Diagnosis Penyakit Asma dan Gangguan Pernapasan (Studi Kasus Rumah Sakit Dokter Soetomo) [9].

Penelitian ini merancang sebuah aplikasi sistem pakar yang nantinya akan digunakan untuk membantu dalam melakukan diagnose penyakit asma dan gangguan pernapasan pada Rumah Sakit Dokter Soetomo. Dengan adanya aplikasi ini nantinya diharapkan dapat membantu anggota dalam mendapatkan informasi seputar penyakit asma dan gangguan pernapasan

dan dapat berkonsultasi langsung dengan pakar tanpa harus bertatap muka, sehingga diagnose penyakit dapat dilakukan sedini mungkin.

Dari enam jurnal yang sudah dijelaskan diatas, dapat dibuat tabel sebagai berikut:

Tabel 2.1 Penelitian Terkait

No	Judul	Masalah	Metode	Hasil
1.	Analisa Kelayakan Truk Pengangkut Material Alam PT. Arga Wastu Sluke – Rembang Menggunakan <i>Fuzzy Logic</i> Tsukamoto	Banyak truk yang mengalami kerusakan diluar rata-rata yang sudah diperkirakan oleh perusahaan, kondisi truk tidak layak jalan membuat truk semakin rusak karena dipaksa untuk beroperasi sehingga sering terjadi kecelakaan.	<i>Fuzzy Inference</i> (FIS) <i>Tsukamoto</i>	Sebuah sistem untuk analisa kelayakan truk pengangkut material alam PT. Arga Wastu
2.	Sistem Pakar Untuk Diagnosa Tingkat Resiko Penyakit Rabies Pada Anjing Menggunakan Metode <i>Fuzzy Inference System</i> (FIS) Tsukamoto	Rabies merupakan penyakit yang sangat berbahaya, anjing merupakan salah satu hewan yang mengidap rabies, sedangkan anjing yang mengidap rabies cenderung menyerang apa saja yang ada di dekatnya. Melalui gigitan itulah penyakit rabies bisa menular pada manusia. Tetapi banyak masyarakat tidak menyadari	<i>Fuzzy Inference</i> (FIS) <i>Tsukamoto</i>	Sebuah aplikasi untuk mendiagnosa tingkat resiko penyakit rabies berdasarkan gejala klinis yang timbul.

		tentang penyakit rabies pada anjing.		
3.	Klasifikasi Penerimaan Beasiswa Dengan Menggunakan Logika Fuzzy Tsukamoto (Studi Kasus Politeknik Kesehatan Kementrian Kesehatan Semarang)	Beasiswa harus diberikan kepada penerima yang layak dan pantas untuk mendapatkannya. Tetapi dalam melakukan seleksi terdapat sebuah kendala, karena banyaknya pelamar beasiswa dan banyaknya kriteria yang digunakan untuk menentukan keputusan penerima beasiswa yang sesuai dengan harapan.	<i>Fuzzy Inference (FIS)</i> <i>Tsukamoto</i>	Sebuah aplikasi yang bisa digunakan untuk membantu dalam pengambilan keputusan untuk penilaian penerimaan beasiswa pada politeknik kesehatan kementrian kesehatan Semarang dengan menggunakan logika <i>fuzzy tsukamoto</i> .
4.	Implementasi Logika Fuzzy Tsukamoto Dalam Menentukan Harga Mobil Toyota Avanza 1.3 G M/T Bekas	Jula beli mobil merupakan suatu kegiatan transaksi yang mungkin sering ditemukan dalam kehidupan sehari-hari. Tetapi dalam pembelian mobil bekas masih banyak pembeli mengalami kerugian saat membeli mobil bekas, hal ini disebabkan oleh kurang memperhatikan faktor-faktor yang mempengaruhi	<i>Fuzzy Inference (FIS)</i> <i>Tsukamoto</i>	Sebuah aplikasi yang mampu membantu masyarakat dalam menentukan harga jual dan beli mobil Toyota Aavanza 1.3 G M/T bekas pada saat akan melakukan transaksi jual beli.

		harga mobil bekas.		
5.	Sistem Diagnosa Level Asma Menggunakan <i>Fuzzy Inference System</i>	Asma mempunyai tingkat fatalitas yang rendah namun kasusnya cukup banyak di negara dengan pendapatan menengah kebawah. Apabila tidak dicegah dan ditangani dengan baik, maka diperkirakan akan terjadi peningkatan prevalensi di masa yang akan datang. Pengetahuan mengenai asma sangat penting dalam mencapai kontrol asma.	<i>Fuzzy Inference (FIS)</i> Mandani	Sebuah sistem untuk membantu Rumah Sakit Pertamina Cilacap dalam diagnosa level penyakit asma.
6.	Rancang Bangun Aplikasi Sistem Pakar Untuk Diagnosis Penyakit Asma dan Gangguan Pernapasan (Studi Kasus Rumah Sakit Dokter Soetomo)	Banyaknya pasien yang datang untuk berobat dan berkonsultasi mengenai penyakit asma tetapi karena kurangnya dokter ahli atau seorang pakar penyakit asma maka pelayanan kurang maksimal. Selain itu belum adanya aplikasi sistem yang dapat membantu dokter	<i>Forward Chaining</i>	Sistem pakar untuk diagnosis penyakit asma dan gangguan pernapasan serta terdapat solusi yang bisa membantu pasien.

		di Poli Paru RSUD Dr. Soetomo, maka untuk menghindari kekeliruan mengenai penyakit asma tersebut maka dibutuhkan aplikasi sistem pakar untuk membantu kerja dari dokter di rumah sakit tersebut		
--	--	---	--	--

2.2 Landasan Teori

2.2.1 Kecerdasan Buatan

Kecerdasan Buatan berasal dari bahasa Inggris “*Artificial Intelligence*” yaitu *intelligence* adalah kata sifat yang berarti cerdas, sedangkan *artificial* artinya buatan. Kecerdasan Buatan yang dimaksud di sini merujuk pada mesin yang mampu berpikir, menimbang tindakan yang akan diambil, dan mampu mengambil keputusan seperti yang dilakukan oleh manusia [3].

2.2.1.1 Definisi Kecerdasan Buatan Menurut Para Ahli

a. Menurut Alan Turing

Alan Turing adalah seorang ahli matematika berkebangsaan Inggris yang dijuluki bapak komputer modern dan pembongkar sandi Nazi dalam era Perang Dunia II 1950, menetapkan definisi *Artificial Intelligence* : “Jika komputer tidak dapat dibedakan dengan manusia saat berbincang melalui terminal komputer, maka bisa dikatakan komputer itu cerdas, mempunyai kecerdasan”[3].

b. Herbert Alexander Simon (June 15, 1916-16-February 9, 2001):

“Kecerdasan buatan (*artificial intelligence*) merupakan kawasan penelitian, aplikasi, dan intruksi yang terkait dengan program komputer

untuk melakukan sesuatu hal yang dalam pandangan manusia adalah cerdas”[3].

c. Menurut John McCarthy

John Mccharty dari Stanford mendefinisikan kecerdasan buatan sebagai “Kemampuan untuk mencapai sukses dalam menyelesaikan suatu permasalahan”[3].

d. Menurut Rich Knight

“Kecerdasan Buatan merupakan sebuah studi tentang bagaimana membuat komputer melakukan hal-hal yang pada saat ini dapat dilakukan lebih baik oleh manusia”[3].

2.2.1.2 Tujuan Kecerdasan Buatan

Menurut Winston dan Prendegast (1984), tujuan dari kecerdasan buatan adalah:

- a. Membuat mesin menjadi lebih pintar (tujuan utama)
- b. Memahami apa itu kecerdasan (tujuan ilmiah)
- c. Membuat mesin lebih bermanfaat (tujuan *enterpreneurial*) [3].

2.2.2 Pengertian Sistem Pakar

Sistem pakar adalah suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu masalah. Sistem pakar akan memberikan pemecahan suatu masalah yang didapat dari dialog dengan pengguna. Dengan bantuan sistem pakar seseorang yang bukan pakar/ahli dapat menjawab pertanyaan, menyelesaikan masalah serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar. Istilah sistem pakar berasal dari istilah *knowledge-based expert system*. Istilah ini muncul karena untuk memecahkan masalah, sistem pakar menggunakan pengetahuan seorang pakar yang dimasukkan ke dalam komputer. Seorang yang bukan pakar menggunakan sistem pakar untuk meningkatkan kemampuan pemecahan masalah, sedangkan seorang pakar menggunakan sistem pakar untuk *knowledge assistant* [3].

2.2.2.1 Definisi Sistem Pakar Menurut Para Ahli:

1. Turban (2001, p402)

“Sistem pakar adalah sebuah sistem yang menggunakan pengetahuan tersebut dimasukkan ke dalam sebuah komputer dan kemudian digunakan untuk menyelesaikan masalah-masalah yang biasanya membutuhkan kepakaran atau keahlian manusia”[3].

2. Jackson (1999, p3)

“Sistem pakar adalah program komputer yang merepresentasikan dan melakukan penalaran dengan pengetahuan beberapa pakar untuk memecahkan masalah atau memberikan saran”[3].

3. Luger dan Stubblefield (1993, p308)

“Sistem pakar adalah program yang berisikan pengetahuan yang menyediakan solusi ‘kualitas pakar’ kepada masalah-masalah dalam bidang (domain) yang spesifik”[3].

2.2.2.2 Manfaat Sistem Pakar

Manfaat sistem pakar antara lain adalah:

1. Meningkatkan produktivitas, karena sistem pakar dapat bekerja lebih cepat daripada manusia.
2. Membuat seorang yang awam bekerja seperti layaknya seorang pakar.
3. Meningkatkan kualitas, dengan member nasehat yang konsisten dan mengurangi kesalahan.
4. Mampu menangkap pengetahuan dan kepakaran seseorang.
5. Dapat beroperasi di lingkungan yang berbahaya.
6. Memudahkan akses pengetahuan seorang pakar.
7. Andal. Sistem pakar tidak pernah menjadi bosan dan kelelahan atau sakit.
8. Meningkatkan kapabilitas sistem komputer. Integrasi sistem pakar dengan sistem komputer lain membuat sistem lebih efektif dan mencakup lebih banyak aplikasi.

9. Mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti. Berbeda dengan sistem komputer konvensional, sistem pakar dapat bekerja dengan informasi yang tidak lengkap. Pengguna dapat merespons dengan: “tidak tahu” atau “tidak yakin” pada satu atau lebih pertanyaan selama konsultasi dan sistem pakar tetap akan memberikan jawabannya.
10. Bisa digunakan sebagai media pelengkap dalam pelatihan. Pengguna pemula yang bekerja dengan sistem pakar akan menjadi lebih berpengalaman karena adanya fasilitas penjelas yang berfungsi sebagai guru.
11. Meningkatkan kemampuan untuk menyelesaikan masalah karena sistem pakar mengambil sumber pengetahuan dari banyak pakar.[3]

2.2.2.3 Kekurangan Sistem Pakar

Kekurangan Sistem Pakar antara lain adalah:

1. Biaya yang sangat mahal untuk membuat dan memeliharanya.
2. Sulit dikembangkan karena keterbatasan keahlian dan ketersediaan pakar.
3. Sistem pakar tidak 100% bernilai benar [3].

2.2.2.4 Ciri-ciri Sistem Pakar

Ciri-ciri dari Sistem Pakar adalah sebagai berikut:

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 /*rule* tertentu.
5. Mudah dimodifikasi.
6. Basis pengetahuan dan mekanisme inferensi terpisah.
7. Keluarannya bersifat anjuran.

8. Sistem dapat mengaktifkan kaidah secara searah yang sesuai, dituntun oleh dialog pengguna [3].

2.2.2.5 Konsep Dasar Sistem Pakar

Konsep dasar sistem pakar meliputi enam hal berikut ini.

1. Kepakaran (*Expertise*)

Kepakaran merupakan suatu pengetahuan yang diperoleh dari pelatihan, membaca, dan pengalaman. Kepakaran inilah yang memungkinkan para ahli dapat mengambil keputusan lebih cepat dan lebih baik daripada seseorang yang bukan pakar. Kepakaran itu sendiri meliputi pengetahuan tentang:

- a. Fakta-fakta tentang bidang permasalahan tertentu.
- b. Teori-teori tentang bidang permasalahan tertentu.
- c. Aturan-aturan dan prosedur-prosedur menurut bidang permasalahan umumnya.
- d. Aturan *heuristic* yang harus dikerjakan dalam suatu situasi tertentu.
- e. Strategi global untuk memecahkan permasalahan.
- f. Pengetahuan tentang pengetahuan (*meta knowledge*) [3].

2. Pakar (*expert*)

Pakar adalah seseorang yang mempunyai pengetahuan, pengalaman, dan metode khusus, serta mampu menerapkannya untuk memecahkan masalah atau memberi nasihat. Seorang pakar harus mampu menjelaskan dan mempelajari hal-hal baru yang berkaitan dengan topik permasalahan, jika perlu harus mampu menyusun kembali pengetahuan-pengetahuan yang didapatkan, dan dapat memecahkan aturan-aturan serta menentukan relevansi kepakarannya. Jadi seorang pakar harus mampu melakukan kegiatan-kegiatan berikut:

- a. Mengenali dan memformulasikan permasalahan.
- b. Memecahkan permasalahan secara cepat dan tepat.
- c. Menerangkan pemecahannya.

- d. Belajar dari pengalaman.
- e. Merestrukturasi pengetahuan
- f. Memecahkan aturan-aturan
- g. Menentukan relevansi [3].

3. Pemindahan Kepakaran (*Transferring Expertise*)

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:

- a. Akuisisi pengetahuan (dari pakar atau sumber lain)
- b. Representasi pengetahuan (pada komputer)
- c. Inferensi pengetahuan
- d. Pemindahan pengetahuan ke pengguna [3].

4. Inferensi (*Inferencing*)

Inferensi adalah sebuah prosedur (program) yang mempunyai kemampuan dalam melakukan penalaran. Inferensi ditampilkan pada suatu komponen yang disebut mesin inferensi yang mencakup prosedur-prosedur mengenai pemecahan masalah. Semua pengetahuan yang dimiliki oleh seorang pakar disimpan pada basis pengetahuan oleh sistem pakar. Tugas mesin inferensi adalah mengambil kesimpulan berdasarkan basis pengetahuan yang dimilikinya [3].

5. Aturan-aturan (*Rule*)

Kebanyakan *software* sistem pakar komersial adalah sistem yang berbasis *rule* (*rule-based system*), yaitu pengetahuan disimpan terutama dalam bentuk *rule*, sebagai prosedur-prosedur pemecahan masalah [3].

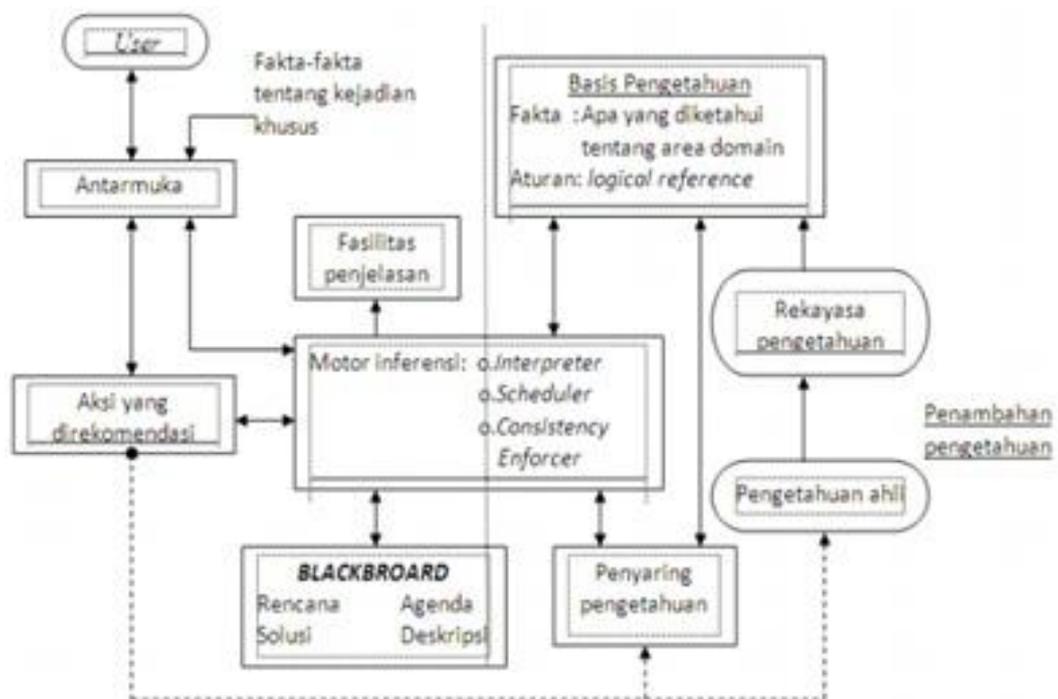
6. Kemampuan menjelaskan (*Explanation Capability*)

Fasilitas lain dari sistem pakar adalah kemampuannya untuk menjelaskan saran atau rekomendasi yang diberikannya. Penjelasan dilakukan dalam subsistem yang disebut subsistem

penjelasan (*explanation*). Bagian dari sistem ini memungkinkan sistem untuk memeriksa penalaran yang dibuatnya sendiri dan menjelaskan operasi-operasinya [3].

2.2.2.6 Struktur Sistem Pakar

Ada dua bagian penting dari sistem pakar, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*). Lingkungan pengembangan digunakan oleh pembuat sistem pakar untuk membangun komponen-komponennya dan memperkenalkan pengetahuan ke dalam *knowledge base* (basis pengetahuan). Lingkungan konsultasi digunakan oleh pengguna untuk berkonsultasi sehingga pengguna mendapatkan pengetahuan dan nasihat dari sistem pakar layaknya berkonsultasi dengan seorang pakar. Gambar 2.1 menunjukkan komponen-komponen yang penting dalam sebuah sistem pakar [3].



Gambar 2.1 Komponen-komponen dalam sebuah sistem pakar [3]

Keterangan:

1. Akuisisi Pengetahuan

Subsistem ini digunakan untuk memasukkan pengetahuan dari seorang pakar dengan cara merekayasa pengetahuan agar bisa diproses oleh komputer dan menaruhnya ke dalam basis pengetahuan dengan format tertentu (dalam bentuk representasi pengetahuan). Sumber-sumber pengetahuan bisa diperoleh dari pakar, buku, dokumen multimedia, basis data, laporan riset khusus, dan informasi yang terdapat di Web [3].

2. Basis Pengetahuan (*Knowledge Base*)

Basis pengetahuan mengandung pengetahuan yang diperlukan untuk memahami, memformulasikan, dan menyelesaikan masalah. Basis pengetahuan terdiri dari dua elemen dasar, yaitu:

- a. Fakta, misalnya situasi, kondisi, atau permasalahan yang ada.
- b. Rule (Aturan), untuk mengarahkan penggunaan pengetahuan dalam memecahkan masalah [3].

3. Mesin Inferensi (*Inference Engine*)

Mesin inferensi adalah sebuah program yang berfungsi untuk memandu proses penalaran terhadap suatu kondisi berdasarkan pada basis pengetahuan yang ada, memanipulasi dan mengarahkan kaidah, model, dan fakta yang disimpan dalam basis pengetahuan untuk mencapai solusi atau kesimpulan. Dalam prosesnya, mesin inferensi menggunakan strategi pengendalian, yaitu strategi yang berfungsi sebagai panduan arah dalam melakukan proses penalaran. Ada tiga teknik pengendalian yang digunakan, yaitu dalam melakukan proses penalaran. Ada tiga teknik pengendalian yang digunakan, yaitu *forward chaining*, *backward chaining*, dan gabungan dari kedua teknik tersebut [3].

4. Daerah Kerja (*Blackboard*)

Untuk merekan hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah masalah yang sedang

terjadi, sistem pakar membutuhkan *Blackboard*, yaitu area pada memori yang berfungsi sebagai basis data. Tiga tipe keputusan yang dapat direkam pada *blackboard*, yaitu:

- a. Rencana : bagaimana menghadapi masalah
- b. Agenda : aksi-aksi potensial yang sedang menunggu untuk dieksekusi
- c. Solusi : calon aksi yang akan dibangkitkan [3].

5. Antarmuka Pengguna (*User Interface*)

Digunakan sebagai media komunikasi antara pengguna dan sistem pakar. Komunikasi ini paling bagus bila disajikan dalam bahasa alami (*natural language*) dan dilengkapi dengan grafik, menu, dan formulir elektronik. Pada bagian ini akan terjadi dialog antara sistem pakar dan pengguna [3].

6. Subsistem Penjelasan (*Explanation Subsystem / justifier*)

Berfungsi member penjelasan kepada pengguna, bagaimana suatu kesimpulan dapat diambil. Kemampuan seperti ini sangat penting bagi pengguna untuk mengetahui proses pemindahan keahlian pakar maupun dalam pemecahan masalah [3].

7. Sistem Perbaikan Pengetahuan (*Knowledge Refining System*)

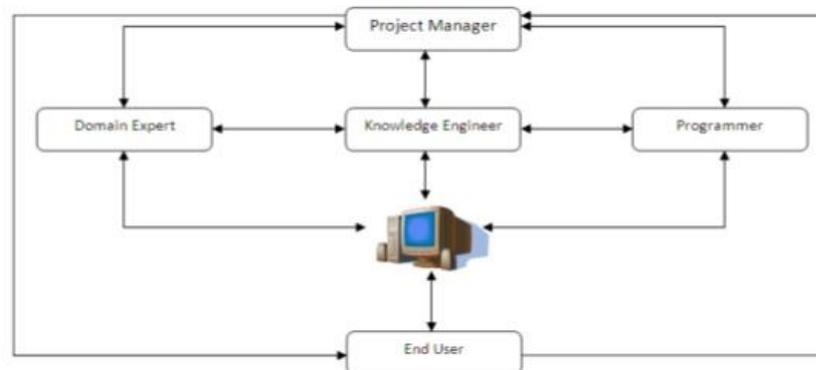
Kemampuan memperbaiki pengetahuan (*knowledge refining system*) dari seorang pakar diperlukan untuk menganalisis pengetahuan, belajar dari kesalahan masa lalu, kemudian memperbaiki pengetahuannya sehingga dapat dipakai pada masa mendatang. Kemampuan evaluasi diri seperti itu diperlukan oleh program agar dapat menganalisis alasan-alasan kesuksesan dan kegagalannya dalam mengambil kesimpulan. Dengan cara ini basis pengetahuan yang lebih baik dan penalaran yang lebih efektif akan dihasilkan [3].

8. Pengguna (*User*)

Pada umumnya pengguna sistem pakar bukanlah seorang pakar (*non-expert*) yang membutuhkan solusi, saran, atau pelatihan (*training*) dari berbagai permasalahan yang ada [3].

2.2.2.7 Tim Pengembangan Sistem Pakar.

Tim Pengembang Sistem Pakar



Gambar 2.2 Tim pengembangan sistem pakar [3].

Keterangan:

Domain expert adalah pengetahuan dan kemampuan seseorang pakar untuk menyelesaikan masalah terbatas pada keahliannya saja. Misalnya seseorang pakar penyakit jantung, ia hanya mampu menangani masalah-masalah yang berkaitan dengan penyakit jantung saja. Ia tidak bisa menyelesaikan masalah-masalah ekonomi, politik, hokum, dan lain-lain. Keahlian inilah yang dimasukkan dalam sistem pakar.

Knowledge Engineer (Perekayasa Pengetahuan) adalah orang yang mampu mendesain, membangun, dan menguji sebuah pakar.

Programer adalah orang yang membuat program sistem pakar, mengode domain pengetahuan agar dapat dimengerti oleh computer.

Project manager adalah pemimpin dalam tim pengembang sistem pakar.

End-User (biasanya disebut user saja) adalah orang yang menggunakan sistem pakar.[3]

2.2.2.8 Rule Sebagai Teknik Representasi Pengetahuan

Setiap *rule* terdiri dari dua bagian, yaitu IF disebut *evidence* (fakta-fakta) dan bagian THEN disebut Hipotesis atau kesimpulan.

Syntax rule adalah:

IF E THEN H

E : *evidence* (fakta-fakta) yang ada

H : Hipotesis atau kesimpulan yang dihasilkan

Secara umum, *rule* mempunyai *evidence* lebih dari satu yang dihubungkan oleh kata penghubung AND atau OR, atau kombinasi keduanya. Tetapi biasakan menghindari penggunaan AND dan OR secara sekaligus dalam satu *rule*.

IF (E₁ AND E₂ AND E₃ AND E_n) THEN H

IF (E₁ OR E₂ OR E₃ OR E_n) THEN H

Satu *evidence* bisa juga mempunyai hipotesis lebih dari satu.

IF E THEN (H₁ AND H₂ AND H₃..... AND H_n). [3]

2.2.3 Logika Fuzzy

Konsep tentang logika *fuzzy* diperkenalkan oleh Prof. Lotfi Astor Zadeh pada 1962. Logika *fuzzy* adalah metodologi sistem kontrol pemecahan masalah, yang cocok untuk diimplementasikan pada sistem, mulai dari sistem yang sederhana, sistem kecil, *embedded system*, jaringan PC, *multi-chanel* atau *workstation* berbasis akuisisi data, dan sistem kontrol. Metodologi ini dapat diterapkan pada perangkat keras, perangkat lunak atau kombinasi keduanya. Dalam logika klasik dinyatakan bahwa

segala sesuatu bersifat biner, yang artinya adalah hanya mempunyai dua kemungkinan, “Ya atau Tidak”, “Benar atau Salah”, “Baik atau Buruk”, dan lain-lain. Oleh karena itu, semua ini dapat mempunyai nilai keanggotaan 0 atau 1. Akan tetapi, dalam logika *fuzzy* memungkinkan nilai keanggotaan berada diantara 0 dan 1. Artinya, bisa saja suatu keadaan mempunyai dua nilai “Ya dan Tidak”, “Benar dan Salah”, “Baik dan Buruk” secara bersamaan, namun besar nilainya tergantung pada bobot keanggotaan yang dimilikinya. Logika *fuzzy* dapat digunakan di berbagai bidang, seperti pada sistem diagnosis penyakit (dalam bidang kedokteran); pemodelan sistem pemasaran, riset operasi (dalam bidang ekonomi); kendali kualitas air, prediksi adanya gempa bumi, klasifikasi dan pencocokan pola (dalam bidang teknik) [3].

2.2.3.1 Dasar - Dasar Logika *Fuzzy*

Untuk memahami logika *fuzzy*, sebelumnya perhatikan dahulu tentang konsep himpunan *fuzzy*. Himpunan *fuzzy* memiliki 2 atribut, yaitu:

- a. Linguistik, yaitu nama suatu kelompok yang mewakili suatu keadaan tertentu dengan menggunakan bahasa alami, misalnya DINGIN, SEJUK, PANAS mewakili variabel temperature. Contoh lain misalnya MUDA, PAROBAYA, TUA, mewakili variabel umur.[3]
- b. Numeris, yaitu suatu nilai yang menunjukkan ukuran dari suatu variabel, misalnya 10, 35, 40, dan sebagainya.[3]

Disamping itu, ada beberapa hal yang harus dipahami dalam memahami logika *fuzzy*, yaitu:

- a. Variabel *fuzzy*, yaitu variabel yang akan dibahas dalam suatu sistem *fuzzy*.
Contoh: penghasilan, temperature, permintaan, umur, dan sebagainya.[3]

- b. Himpunan *fuzzy* yaitu suatu kelompok yang mewakili suatu keadaan tertentu dalam suatu variabel *fuzzy*.
- c. Semesta pembicaraan, yaitu seluruh nilai yang diizinkan untuk dioperasikan dalam suatu variabel *fuzzy*. [3]

Contoh: [3]

Semesta pembicaraan untuk variabel permintaan: $[0 + \infty]$

Semesta pembicaraan untuk variabel temperatur: $[-10 90]$

- d. Domain himpunan *fuzzy*, yaitu seluruh nilai yang diizinkan dalam semesta pembicaraan dan boleh dioperasikan dalam suatu himpunan *fuzzy*. Pada Gambar 2.2 di atas domain untuk himpunan TURUN dan himpunan NAIK masing-masing adalah: [3]

Domain himpunan TURUN = $[0 5000]$

Domain himpunan NAIK = $[1000 +\infty]$

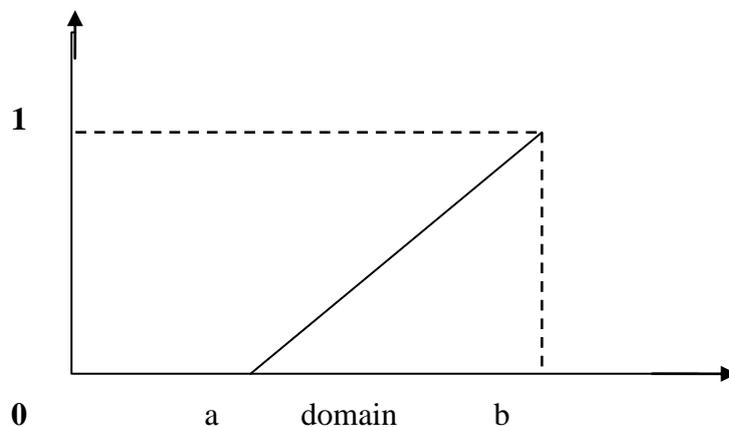
2.2.3.2 Fungsi Keanggotaan

Fungsi keanggotaan adalah grafik yang mewakili besar dari derajat keanggotaan masing-masing variabel input yang berbeda dalam interval antara 0 dan 1. Derajat keanggotaan sebuah variabel x dilambangkan dengan simbol $\mu(x)$. Rule – rule menggunakan nilai keanggotaan sebagai faktor bobot untuk menentukan pengaruhnya pada saat melakukan inferensi untuk menarik kesimpulan [3].

2.2.3.3 Grafik Keanggotaan Fungsi Linear

Pada grafik keanggotaan fungsi linear, sebuah variabel input dipetakan ke derajat keanggotaannya dengan digambarkan sebagai suatu garis lurus.

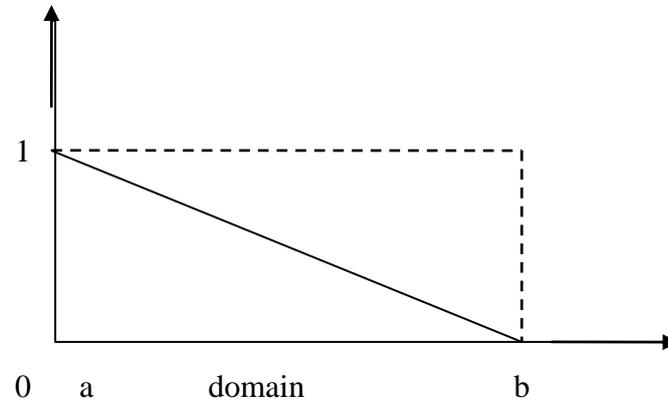
Ada dua grafik keanggotaan linear. Pertama, grafik keanggotaan kurva linear naik, yaitu kenaikan himpunan *fuzzy* dimulai pada nilai dominan yang memiliki derajat keanggotaan nol $[0]$ bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi [3].



Gambar 2.3 Grafik keanggotaan kurva linear naik

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

Kedua, grafik keanggotaan kurva linear turun, yaitu himpunan *fuzzy* dimulai dari nilai dominan dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai dominan yang memiliki derajat keanggotaan lebih rendah.[3]

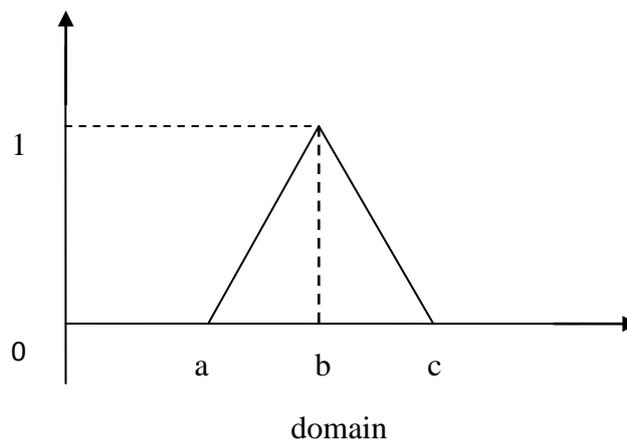


Gambar 2.4 Grafik keanggotaan kurva linear turun

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

2.2.3.4 Grafik Keanggotaan Kurva Segitiga

Grafik keanggotaan kurva segitiga pada dasarnya merupakan gabungan antara dua garis (linear).

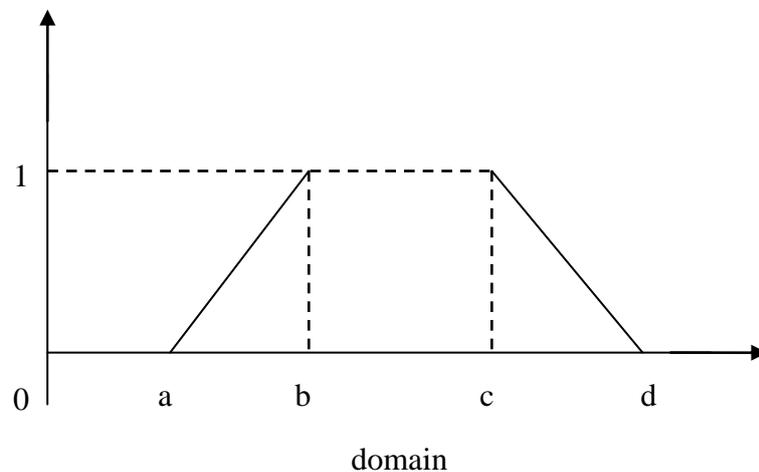


Gambar 2.5 Grafik keanggotaan kurva segitiga

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

2.2.3.5 Grafik Keanggotaan Kurva Trapesium

Grafik keanggotaan kurva trapesium pada dasarnya seperti bentuk segitiga, hanya saja ada beberapa titik yang memiliki nilai keanggotaan 1.



Gambar 2.6 Grafik keanggotaan kurva trapesium

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

2.2.3.6 Operasi Himpunan Fuzzy

Operasi himpunan fuzzy diperlukan untuk untuk proses inferensi atau penalaran. Dalam hal ini yang dioperasikan adalah derajat keanggotaan. Derajat keanggotaan sebagai hasil dari oprasi dua buah himpunan fuzzy disebut sebagai fire strength atau α -predikat.

Berikut beberapa operasi dasar yang paling sering digunakan untuk mengkombinasi dan memodifikasi himpunan *fuzzy* [3].

a. Operasi gabungan(union)

Operasi gabungan (sering disebut operasi OR) dari himpunan fuzzy A dan B dinyatakan sebagai $A \cup B$. dalam sistem logika fuzzy, operasi gabungan disebut sebagai Max. Operasi Max ditulis dengan persamaan berikut.

$$\mu_{A \cup B}(x) = \max. \{\mu_A(x), \mu_B(x)\} \text{ untuk setiap } x \in X$$

Derajat Keanggotaan setiap unsur himpunan fuzzy $A \cup B$ adalah derajat keanggotaannya pada himpunan fuzzy A atau B yang memiliki nilai terbesar.[3]

b. Operasi Irisan (intersection)

Operasi Irisan (sering disebut operator AND) dari himpunan fuzzy A dan B dinyatakan sebagai $A \cap B$. Dalam sistem logika fuzzy, operasi irisan disebut sebagai Min. Operasi Min ditulis dengan persamaan sebagai berikut

$$\mu_{A \cap B}(x) = \min\{\mu_A(x), \mu_B(x)\} \text{ untuk setiap } x \in X$$

Derajat keanggotaan setiap unsur himpunan fuzzy $A \cap B$ adalah derajat keanggotannya pada himpunan fuzzy A dan B yang memiliki nilai terkecil. [3]

c. Operai Komplemen (Complement)

Bila himpunan fuzzy A pada himpunan universal X mempunyai fungsi keanggotaan $\mu_A(x)$ maka komponen dari himpunan fuzzy A (sering disebut NOT) adalah himpunan fuzzy A^c dengan fungsi keanggotaan untuk setiap x elemen X.[3]

2.2.3.7 Penalaran Monoton

Penalaran Monoton digunakan untuk merealisasikan himpunan fuzzy A pada variabel x dan himpunan fuzzy B pada variable y dengan cara membuat implikasi berikut [3]

IF x is A THEN y is B

2.2.3.8 Fungsi Implikasi

Dalam basis pengetahuan himpunan fuzzy, tiap-tipa rule selalu berhubungan dengan relasi fuzzy. Dalam fungsi implikasi, biasanya digunakan bentuk berikut [3]

IF x is A THEN y is B

Dengan x dan y adalah saklar, sedangkan himpunan A dan B adalah himpunan fuzzy. Proporsi setelah IF disebut sebagai anteseden, sedangkan proporsi THEN disebut sebagai konsekuen. Dengan menggunakan operator fuzzy, proporsi ini dapat diperluas sebagai berikut [3]

$IF(x_1 \text{ is } A_1) \cdot$

Dengan tanda (titik) adalah operator OR atau AND. Secara umum, ada 2 fungsi implikasi yang dapat digunakan, yaitu :

a. Min (minimum)

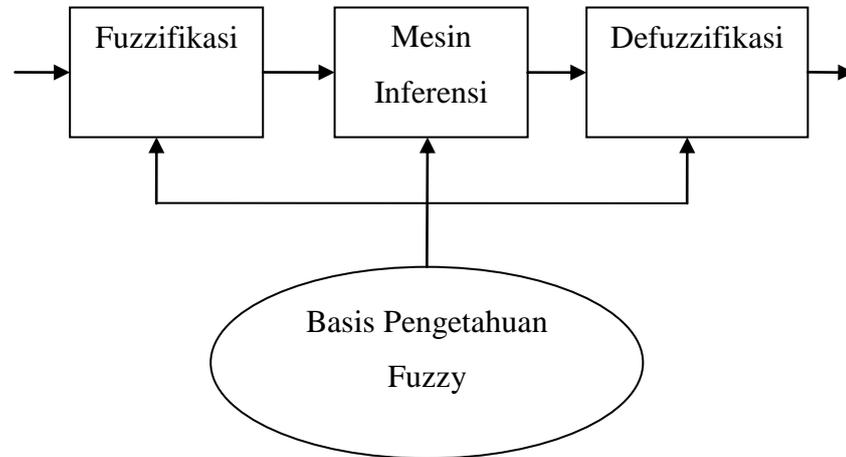
Fungsi ini digunakan untuk mendapatkan nilai α - predikat hasil implikasi dengan cara memotong output himpunan fuzzy sesuai dengan derajat keanggotaan yang kecil [3].

b. Dot (product)

Fungsi ini digunakan untuk mendapatkan nilai α -predikat hasil implikasi dengan cara menskala output himpunan fuzzy sesuai dengan derajat keanggotaan yang terkecil [3].

2.2.3.9 Cara Kerja Logika Fuzzy

Untuk memahami cara kerja logika *fuzzy*, perhatikan struktur elemen dasar sistem *inferensi fuzzy* berikut [3]



Gambar 2.7 Struktur sistem inferensi fuzzy [3]

Keterangan:

Basis Pengetahuan Fuzzy: kumpulan rule-rule *Fuzzy* dalam bentuk pernyataan IF...THEN [3].

Fuzzyfikasi: proses untuk mengubah input sistem yang mempunyai nilai tegas menjadi variabel linguistic menggunakan fungsi keanggotaan yang disimpan dalam basis pengetahuan fuzzy [3].

Mesin Inferensi: proses untuk mengubah input fuzzy menjadi output fuzzy dengan cara mengikuti aturan aturan (*IF-THEN Rules*) yang telah ditetapkan pada basis pengetahuan fuzzy [3].

Defuzzyfikasi: mengubah output fuzzy yang diperoleh dari mesin inferensi menjadi nilai tegas menggunakan fungsi keanggotaan yang sesuai dengan saat dilakukan fuzzyfikasi [3].

Cara kerja logika fuzzy meliputi beberapa tahapan berikut:

1. Fuzzyfikasi [3]
2. Pembentukan basis pengetahuan fuzzy (Rule dalam bentuk IF...THEN) [3]
3. Mesin Inferensi (fungsi implikasi Max-Min atau Dot-Product)
4. Defuzzyfikasi

Banyak cara untuk melakukan defuzzyfikasi, diantaranya metode berikut.

- a. Metode Rata-Rata (*Average*) [3]

$$z^* = \frac{\sum \mu_i z_i}{\sum \mu_i}$$

- b. Metode Titik Tengah (*Center Of Area*) [3]

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

2.2.4 Fuzzy Inference System (FIS) Tsukamoto

Fuzzy inference system (FIS) merupakan pengembangan dari *fuzzy logic* yang telah dikenal sebagai salah satu perkembangan bidang ilmu kecerdasan buatan, yang mampu memberikan solusi dengan mengakomodir penggunaan bahasa alami, yang dihasilkan dari sekumpulan pengetahuan yang ditransfer ke dalam perangkat lunak melalui *inferensi fuzzy*, yang selanjutnya memetakan suatu input menjadi output berdasarkan IF-THEN rule yang diberikan [4].

Pada *Fuzzy Inference System* (FIS) dikenal beberapa metode yang telah populer, seperti: metode Tsukamoto, metode Mamdani, dan metode Sugeno. Setiap metode memiliki karakteristik yang berbeda. Pada metode Tsukamoto, setiap konsekuensi 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 diberikan dengan tegas (*crisp*) berdasarkan α -predikat (*fire strength*). Hasil akhirnya diperoleh dengan menggunakan rata-rata terbobot.

Secara umum bentuk model *fuzzy Tsukamoto* adalah:

$$\text{IF}(X \text{ IS } A) \text{ and } (Y \text{ IS } B) \text{ Then } (Z \text{ IS } C) [3]$$

Dimana A, B, dan C adalah himpunan *fuzzy*.

Misalkan diketahui 2 rule berikut.

$$\text{IF } (x \text{ is } A_1) \text{ AND } (y \text{ is } B_1) \text{ THEN } (z \text{ is } C_1)$$

$$\text{IF } (x \text{ is } A_2) \text{ AND } (y \text{ is } B_2) \text{ THEN } (z \text{ is } C_2)$$

Dalam inferensinya, metode Tsukamoto menggunakan tahapan berikut.

1. Fuzzyfikasi [3]
2. Pembentukan basis pengetahuan Fuzzy (Rule dalam bentuk IF...THEN) [3]
3. Mesin Inferensi
Menggunakan fungsi implikasi MIN untuk mendapatkan nilai α -predikat tiap-tiap rule ($\alpha_1, \alpha_2, \alpha_3, \dots, \alpha_n$)[3].
4. Defuzzyfikasi
Menggunakan metode Rata-Rata (*Average*) [3]

$$z^* = \frac{\sum a_i z_1}{\sum a_i}$$

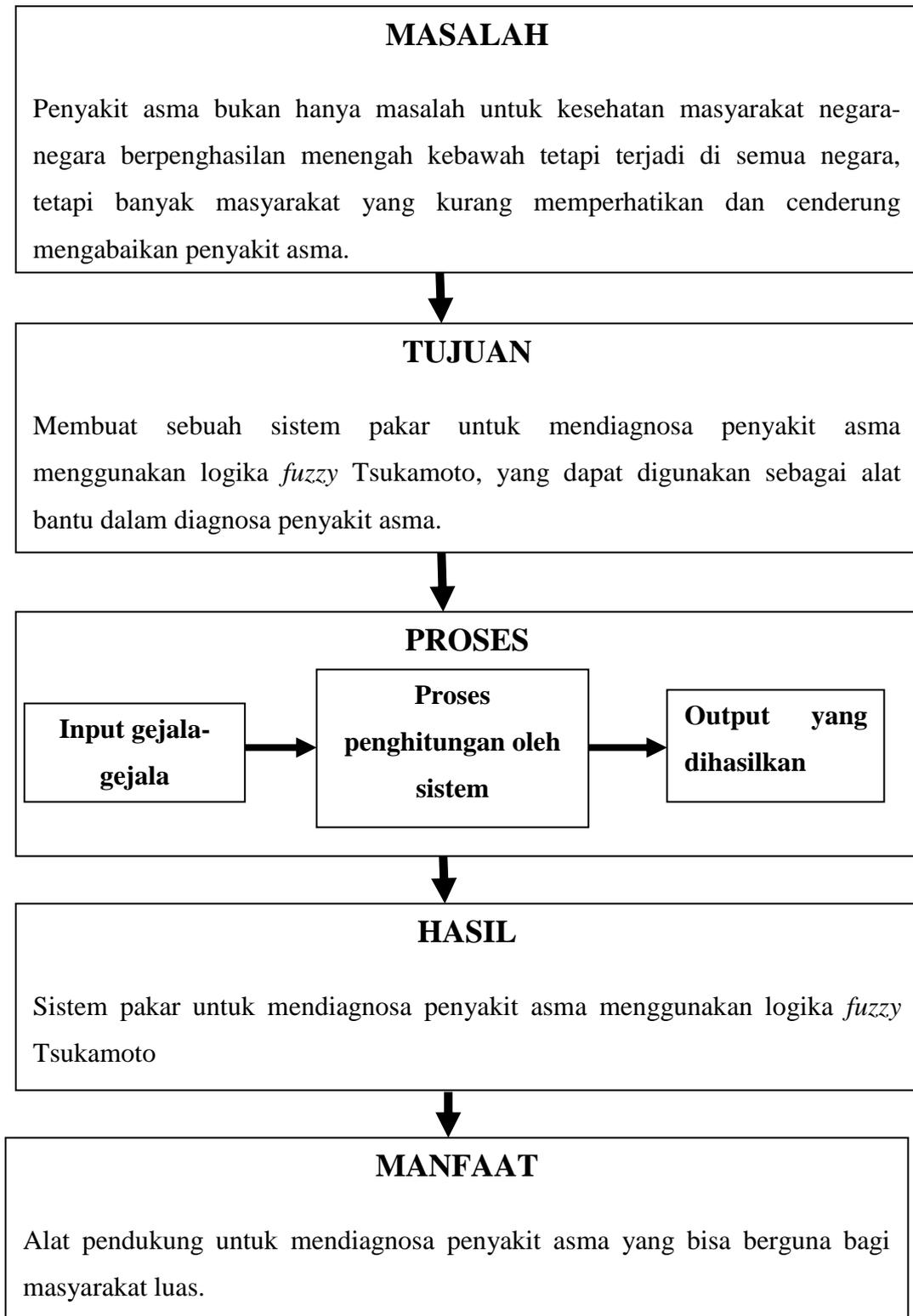
Proses DeFuzzifikasi

Hasil akhir *output* (z) diperoleh dengan menggunakan rata-rata pembobotan:

$$z = \frac{\alpha_1 z_1 + \alpha_2 z_2}{\alpha_1 + \alpha_2}$$

2.3 Kerangka Pemikiran

Berikut merupakan kerangka pemikiran dalam penelitian tentang SISTEM PAKAR DIAGNOSA LEVEL PENYAKIT ASMA MENGGUNAKAN METODE *FUZZY TSUKAMOTO*



Gambar 2.8 Kerangka pemikiran