

BAB 2

LANDASAN TEORI

2.1 Kecerdasan Buatan

Kecerdasan buatan (*artificial intelligent*) merupakan salah satu bagian ilmu komputer yang membuat agar mesin (komputer) dapat melakukan pekerjaan seperti yang sebaik dilakukan oleh manusia. Hal ini dilakukan dengan mempelajari bagaimana manusia berpikir ketika manusia mencoba untuk membuat suatu keputusan dan memecahkan masalah, membagi-bagi proses berpikir tersebut menjadi langkah-langkah dasar dan merancang suatu program komputer yang akan memecahkan masalah dengan mempergunakan langkah-langkah yang sama.

Teknologi kecerdasan buatan dipelajari dalam beberapa bidang, seperti : Robotika dan Sistem Sensor, Penglihatan Komputer (*Computer Vision*), Sistem Saraf Tiruan (*Artificial Neural System*), Agen Cerdas (*Intelligent Agent*) Pengenalan Suara (*Speech Recognition*), Permainan (*Game Playing*) dan Sistem Pakar (*Expert System*).

2.2 Sistem Pakar

Sistem pakar pertama kali dikembangkan oleh komunitas AI pada pertengahan tahun 1960. Sistem pakar yang muncul pertama kali adalah *General Purpose Problem Solver* (GPS) yang dikembangkan oleh Newel & Simon.

Beberapa definisi yang ada untuk sistem pakar (*expert system*) :

- a. Sistem Pakar (*Expert System*) adalah suatu cabang ilmu dari Kecerdasan Buatan (*Artificial Intelligent*) yang digunakan secara luas untuk pengetahuan (keahlian) khusus dalam memecahkan suatu masalah layaknya seorang pakar.
- b. Sistem pakar adalah sistem yang menggunakan pengetahuan manusia (ahli) yang terekam dalam komputer untuk memecahkan persoalan yang biasanya memerlukan keahlian manusia.

Konsep dasar sistem pakar mencakup beberapa persoalan mendasar, antara lain apa yang dimaksud dengan keahlian, siapa yang disebut pakar, bagaimana keahlian dapat ditransfer, dan bagaimana sistem bekerja.

Pakar adalah orang yang memiliki pengetahuan, penilaian, pengalaman, dan metode khusus, serta kemampuan untuk menerapkan bakat ini dalam memberi nasehat dan memecahkan persoalan. Adalah tugas pakar untuk menyediakan pengetahuan tentang bagaimana melaksanakan suatu tugas yang akan dijalankan oleh sistem berbasis pengetahuan. Pengertian lain dari pakar ialah orang yang memiliki keahlian dalam suatu hal, yaitu memiliki pengetahuan atau keahlian khusus yang tidak diketahui dan tidak ada pada kebanyakan orang.

Keahlian adalah pengetahuan ekstensif yang spesifik terhadap tugas yang dimiliki pakar. Tingkat keahlian menentukan performa keputusan. Keahlian sering dicapai dari pelatihan, membaca, dan mempraktikkan. Keahlian mencakup pengetahuan eksplisit, misalnya teori yang dipelajari dari buku teks atau kelas, dan pengetahuan implisit yang diperoleh dari pengalaman.

2.2.1 Manfaat Sistem Pakar

Secara garis besar, ada banyak manfaat bila menggunakan sistem pakar. Beberapa diantaranya yaitu :

1. Memungkinkan orang awam bisa mengerjakan pekerjaan para ahli.

2. Bisa melakukan proses secara berulang secara otomatis.
3. Menyimpan pengetahuan dan keahlian para pakar.
4. Mampu mengambil dan melestarikan keahlian para pakar (terutama yang termasuk keahlian langka)
5. Mampu beroperasi dalam keadaan berbahaya.
6. Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
7. Suatu sistem pakar dapat diberi pengamanan untuk menentukan siapa saja yang mempunyai hak akses untuk menggunakannya dan jawaban yang diberikan oleh sistem terbebas dari proses intimidasi/ancaman, sedangkan seorang pakar bisa saja mendapat ancaman atau tekanan pada saat menyelesaikan masalah.
8. Umumnya kecepatan dalam memecahkan masalah pada suatu sistem pakar relatif lebih cepat dibandingkan oleh seorang pakar manusia.

2.2.2 Kategori Sistem Pakar

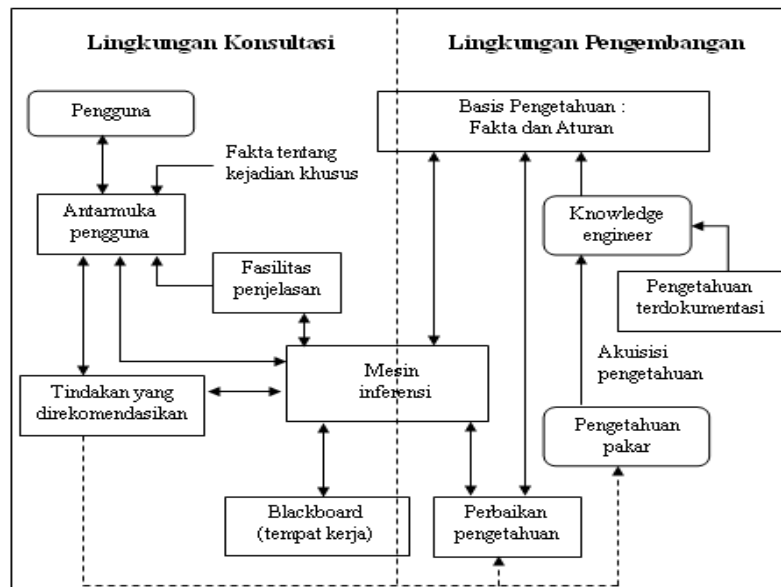
Secara umum sistem pakar dapat diklasifikasikan dalam beberapa area permasalahan, yaitu :

1. **Interpretasi**, yaitu pengambilan keputusan atau deskripsi tingkat tinggi dari sekumpulan data mentah, termasuk diantaranya juga pengawasan, pengenalan ucapan, analisis citra, interpretasi sinyal, dan beberapa analisis kecerdasan.
2. **Prediksi**, yaitu memprediksi akibat-akibat yang dimungkinkan dari situasi-situasi tertentu, diantaranya peramalan, prediksi demografis, peramalan ekonomi, prediksi lalu lintas, estimasi hasil, militer, pemasaran atau peramalan keuangan.
3. **Diagnosis**, yaitu menentukan sebab mal fungsi dalam situasi kompleks yang didasarkan pada gejala-gejala yang teramati, diantaranya medis, elektronis, mekanis dan diagnosis perangkat lunak.

4. **Desain**, yaitu menentukan konfigurasi komponen-komponen sistem yang cocok dengan tujuan- tujuan kinerja tertentu yang memenuhi kendala-kendala tertentu diantaranya layout circuit dan perancangan bangunan.
5. **Perencanaan**, yaitu fokus pada persoalan perencanaan, misalnya pemrograman otomatis. Juga menangani perencanaan jangka pendek dan panjang pada area seperti manajemen proyek, routing, komunikasi, pengembangan produk, aplikasi militer, dan perencanaan keuangan.
6. **Sistem pengawasan**, yaitu membandingkan observasi kelakuan sistem dengan standar yang tampaknya penting untuk keberhasilan pencapaian tujuan. Fitur penting ini berhubungan dengan kecacatan potensial dalam rencana tersebut. Terdapat banyak sistem pengawasan dengan bantuan komputer untuk topik dari kontrol arus lalu lintas udara hingga tugas manajemen fiskal.
7. **Debugging**, yaitu mengandalkan pada kemampuan perencanaan, desain, prediksi untuk membuat spesifikasi atau rekomendasi untuk membetulkan persoalan diagnosis.
8. **Perbaikan**, yaitu mengembangkan dan mengeksekusi rencana untuk mengelola perbaikan persoalan diagnosis tertentu. Sistem tersebut menggabungkan kemampuan debugging, perencanaan, dan eksekusi.
9. **Instruksi**, yaitu mendeteksi dan mengoreksi defisiensi dalam pemahaman domain subjek, diantaranya melakukan instruksi untuk diagnosis, debugging dan perbaikan kerja.
10. **Kontrol**, yaitu secara adaptif mengatur keseluruhan kelakuan sistem. Untuk melakukan ini, sistem kontrol harus berulang kali menginterpretasikan situasi terbaru, memprediksi masa depan, mendiagnosis penyebab persoalan yang terantisipasi, merumuskan rencana pemulihan, dan mengawasi eksekusinya untuk memastikan keberhasilan.

2.2.3 Struktur Sistem Pakar

Sistem pakar dapat ditampilkan dengan dua lingkungan, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) (Gambar 2.1). Lingkungan pengembangan digunakan oleh sistem pakar (ES) builder untuk membangun komponen dan memasukkan pengetahuan ke dalam basis pengetahuan. Lingkungan konsultasi digunakan oleh nonpakar untuk memperoleh pengetahuan dan nasihat pakar. Lingkungan ini dapat dipisahkan setelah sistem lengkap.



Gambar 2.1 Struktur Sistem Pakar

Tiga komponen utama yang tampak secara virtual di setiap sistem pakar adalah **basis pengetahuan, mesin inferensi, dan antarmuka pengguna**. Selain antarmuka pengguna, basis pengetahuan, dan mesin inferensi, dari struktur sistem pakar yang terdapat pada gambar diatas mengandung komponen lain, yaitu akuisisi pengetahuan, blackboard, fasilitas penjelasan, dan perbaikan pengetahuan. (Turban, Efrain, 2005)

2.2.3.1 Antarmuka Pengguna

Antarmuka pemakai merupakan mekanisme yang digunakan oleh pengguna dan sistem pakar untuk berkomunikasi. Pada bagian ini terjadi dialog antara program dan pemakai, yang memungkinkan sistem pakar menerima instruksi dan informasi (input) dari pemakai, juga memberikan informasi (output) kepada pemakai.

2.2.3.2 Basis Pengetahuan

Basis pengetahuan berisi pengetahuan relevan yang diperlukan untuk memahami, merumuskan, dan memecahkan persoalan. Basis tersebut mencakup dua elemen dasar, yaitu :

1. Fakta, misalnya situasi persoalan dan teori area persoalan.
2. Heuristik atau aturan khusus yang mengarahkan penggunaan pengetahuan untuk memecahkan persoalan khusus dalam domain tertentu. Heuristik menyatakan pengetahuan penilaian informasi dalam area aplikasi.

Pada basis pengetahuan terdapat 2 (dua) bentuk pendekatan basis pengetahuan yang umum, yaitu :

1. *Rule-based Reasoning* (Penalaran berbasis pengetahuan).

Pada penalaran berbasis aturan, pengetahuan direpresentasikan dengan menggunakan aturan berbentuk: *IF-THEN*. Bentuk ini digunakan apabila dimiliki sejumlah pengetahuan pakar pada suatu permasalahan tertentu, dan si pakar dapat menyelesaikan masalah tersebut secara berurutan. Disamping itu bentuk ini juga digunakan apabila dibutuhkan penjelasan tentang jejak (langkah-langkah) pencapaian solusi.

2. *Case-based Reasoning* (Penalaran berbasis kasus)

Pada penalaran berbasis kasus, basis pengetahuan akan berisi solusi-solusi yang telah dicapai sebelumnya, kemudian akan diturunkan suatu solusi untuk keadaan yang terjadi sekarang (fakta yang ada). Bentuk ini digunakan apabila *user* menginginkan untuk tahu lebih banyak lagi pada kasus-kasus yang hampir sama

(mirip). Selain itu, bentuk ini juga digunakan apabila telah dimiliki sejumlah situasi atau kasus tertentu dalam basis pengetahuan.

2.2.3.3 Subsistem Akuisisi Pengetahuan

Akuisisi pengetahuan adalah akumulasi, transfer, dan transformasi keahlian pemecahan masalah dari pakar atau sumber pengetahuan terdokumentasi ke program komputer, untuk membangun atau memperluas basis pengetahuan.

2.2.3.4 Mesin Inferensi

Otak sistem pakar adalah mesin inferensi, yang dikenal juga sebagai struktur kontrol atau penerjemah aturan (dalam sistem pakar berbasis aturan). Komponen ini sebenarnya adalah program komputer yang menyediakan metodologi untuk mempertimbangkan informasi dalam basis pengetahuan dan blackboard, dan merumuskan kesimpulan. Komponen ini menyediakan arahan bagaimana menggunakan pengetahuan sistem, yakni dengan mengembangkan agenda yang mengatur dan mengontrol langkah yang diambil untuk memecahkan persoalan kapan pun konsultasi berlangsung.

Terdapat dua pendekatan untuk mengontrol inferensi dalam sistem pakar berbasis aturan, yaitu :

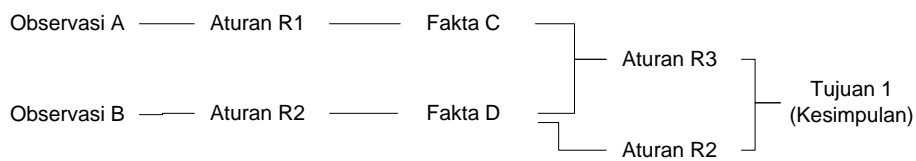
a. Pelacakan ke depan (*forward chaining*)

Pelacakan ke depan adalah pendekatan yang dimotori data (*data driven*). Dalam pendekatan ini pelacakan dimulai dari informasi masukan, dan selanjutnya mencoba menggambarkan kesimpulan. Pelacakan ke depan mencari fakta yang sesuai dengan bagian *IF* dari aturan *IF-THEN*.

Gambar 2.2 Proses Forward Chaining

b. Pelacakan ke belakang (*backward chaining*)

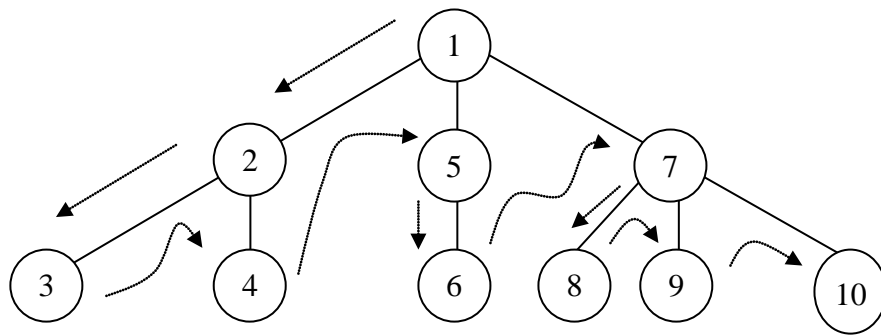
Pelacakan ke belakang adalah pendekatan yang dimotori tujuan (*goal driven*) yaitu pelacakan yang dimulai dari tujuan, selanjutnya dicari aturan yang memiliki tujuan tersebut untuk kesimpulannya. Selanjutnya proses pelacakan menggunakan premis untuk aturan tersebut sebagai tujuan baru dan mencari aturan lain dengan tujuan baru sebagai kesimpulannya.



Gambar 2.3 Proses Backward Chaining

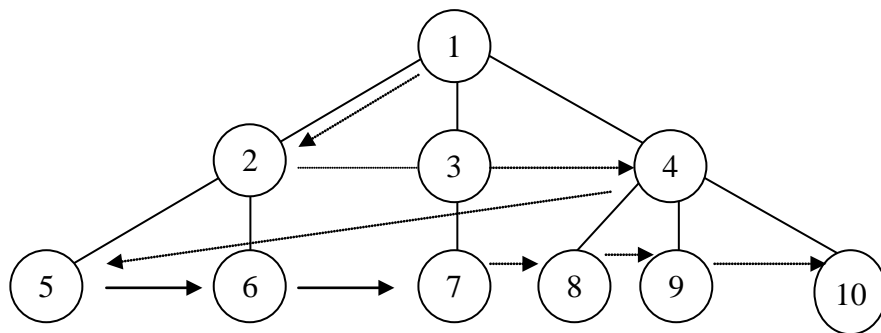
Kedua metode inferensi tersebut dipengaruhi oleh tiga macam penelusuran, yaitu :

- a. *Depth-first search*, melakukan penelusuran kaidah secara mendalam dari simpul akar bergerak menurun ke tingkat yang dalam berurutan.



Gambar 2.4 Diagram Alir Teknik penelusuran Depth-First Search

- b. *Breadth-first search*, bergerak dari simpul akar, simpul yang ada pada setiap tingkat diuji sebelum pindah ke tingkat selanjutnya.



Gambar 2.5 Diagram Alir Teknik penelusuran Breadth-First Search

- c. *Best-first search*, bekerja berdasarkan kombinasi kedua metode sebelumnya.

2.2.3.5 Blackboard (Tempat Kerja)

Blackboard adalah area kerja memori yang disimpan sebagai database untuk deskripsi persoalan terbaru yang ditetapkan oleh data input dan digunakan juga untuk penekanan hipotesis dan keputusan sementara. Tiga tipe keputusan yang dapat direkam dalam blackboard :

- a. Rencana : bagaimana mengatasi persoalan
- b. Agenda : tindakan sebelum dieksekusi
- c. Solusi : hipotesis kandidat dan arah tindakan alternatif yang telah dihasilkan sistem

2.2.3.6 Fasilitas Penjelasan (*Justifier*)

Fasilitas penjelasan untuk komponen tambahan yang akan meningkatkan kemampuan sistem pakar. Komponen ini menggambarkan penalaran sistem kepada pemakai.

2.2.3.7 Perbaikan Pengetahuan

Pakar manusia memiliki sistem perbaikan-pengetahuan, yakni mereka dapat menganalisis pengetahuannya sendiri dan kegunaannya, belajar darinya, dan meningkatkannya untuk konsultasi mendatang. Serupa pula, evaluasi tersebut diperlukan dalam pembelajaran komputer sehingga program dapat menganalisis alasan keberhasilan atau kegagalannya. Hal ini dapat mengarah kepada peningkatan sehingga menghasilkan basis pengetahuan yang lebih akurat dan pertimbangan yang lebih efektif. Komponen tersebut tidak tersedia dalam sistem pakar komersil saat ini, tetapi sedang dikembangkan dalam ES eksperimental pada beberapa universitas dan lembaga riset.

2.3 Representasi Pengetahuan

Dalam pembangunan sistem berbasis pengetahuan, pengetahuan yang telah diekstrak dimasukkan kedalam program komputer oleh proses yang disebut representasi pengetahuan (*knowledge representation*). Sistem representasi pengetahuan merupakan gabungan dari dua elemen yakni struktur data dan prosedur

menafsirkan (*interpretive procedure*) untuk pemakaian pengetahuan yang dimasukkan dalam struktur data

Terdapat empat teknik yang telah dibuktikan efektif untuk representasi pengetahuan, yang paling populer adalah aturan produksi, frame, jaringan semantik, dan script.

2.3.1 Aturan Produksi

Aturan produksi adalah bentuk representasi pengetahuan yang paling populer untuk sistem pakar. Pengetahuan direpresentasikan dalam bentuk :

IF [kondisi] Then [aksi]

Tiap aturan produksi dalam basis pengetahuan mengimplementasikan sebagian keahlian otonomi yang dapat dikembangkan dan dimodifikasi secara terpisah dari aturan lain. Pada saat dimasukkan dan digabungkan kedalam mesin inferensi, set aturan berlaku secara sinergi, memberikan hasil yang lebih baik daripada jumlah hasil aturan individu.

2.3.2 Frame

Frame adalah struktur data yang menyertakan semua pengetahuan tentang objek tertentu. Frame mencakup dua elemen dasar: slot dan facet. Slot adalah atribut yang mendefinisikan objek yang direpresentasikan oleh frame. Tiap slot berisi satu atau lebih facet. Facet mendiskripsikan beberapa pengetahuan atau informasi prosedural tentang atribut dalam slot. Frame menyediakan sarana mengatur pengetahuan dalam slot yang berisi karakter dan atribut

2.3.3 Jaringan Semantik

Jaringan semantik berfokus pada hubungan antar konsep yang berbeda. Jaringan ini adalah gambaran grafis pengetahuan terdiri dari node dan link yang menunjukkan hubungan hierarki antara objek. Jaringan semantik terdiri dari node yang merepresentasikan objek, dan informasi deskriptif tentang objek. Salah satu fakta yang paling menarik tentang jaringan semantik adalah dapat menunjukkan pewarisan. Karena jaringan semantik pada dasarnya berupa hierarki, maka berbagai karakteristik beberapa node sebenarnya mewarisi karakteristik yang lain.

2.3.4 Script

Script merupakan skema representasi pengetahuan yang sama dengan frame. Hanya saja frame menggambarkan objek sedangkan script menggambarkan urutan peristiwa. Sama halnya dengan frame, script juga untuk merepresentasikan situasi atau pengetahuan stereotipe atau pengetahuan yang didasarkan pada karakteristik yang sudah dikenal dan merupakan pengalaman. Berbeda dengan frame, script biasanya direpresentasikan ke dalam konteks tertentu. Penggambaran urutan peristiwa pada script menggunakan serangkaian slot yang berisi informasi tentang orang, objek, dan tindakan-tindakan yang terjadi dalam suatu peristiwa.

Script mempunyai beberapa elemen yang tipikal, yaitu kondisi masukan, prop, role dan scene. Kondisi masukan menggambarkan situasi yang harus dipenuhi sebelum terjadi atau berlaku suatu peristiwa yang ada dalam script. Prop mengacu kepada objek yang digunakan dalam urutan peristiwa yang terjadi. Role mengacu kepada orang-orang yang terlibat dalam script. Hasilnya adalah kondisi yang ada sesudah peristiwa script berlangsung. Track mengacu kepada variasi yang mungkin terjadi dalam script tertentu. Akhirnya, scene menggambarkan urutan peristiwa aktual yang terjadi.

2.4 Faktor Kepastian (*Certainty Factor*)

Teori kepastian bergantung pada penggunaan faktor-faktor kepastian. Faktor kepastian (CF) menyatakan kepercayaan dalam sebuah kejadian (fakta atau hipotesis) berdasarkan bukti (atau penilaian pakar). Ada beberapa metode penggunaan faktor kepastian untuk menangani ketidakpastian di dalam sistem berbasis pengetahuan. Salah satu caranya adalah dengan menggunakan 1,0 atau 100 untuk kebenaran absolut (keyakinan penuh) dan 0 untuk kebohongan pasti. Faktor kepastian bukanlah probabilitas. Sebagai contoh pada saat mengatakan ada 90 persen peluang hujan, maka ada hujan (90 persen) ataupun tidak hujan (10 persen).

Teori kepastian memperkenalkan konsep kepercayaan dan ketidakpercayaan. Konsep ini bebas satu sama lain sehingga tidak dapat dikombinasikan dengan cara yang sama sebagaimana probabilitas, tetapi dapat dikombinasikan menurut persamaan sebagai berikut:

$$CF(P,E) = MB(P,E) - MD(P,E) \dots\dots\dots(2.1)$$

Dimana : CF = faktor kepastian
 MB = ukuran kepercayaan
 MD = ukuran ketidakpercayaan
 P = probabilitas
 E = bukti atau kejadian

2.4.1 Mengkombinasikan Beberapa Faktor Kepastian Dan Satu Aturan.

Anggap aturan dengan operator OR :

IF inflasi rendah, CF = 70 persen; OR
Harga obligasi tinggi, CF = 85 persen;
THEN harga saham akan tinggi

Dari contoh kasus ini, sudah cukup satu *IF* yang benar agar kesimpulannya benar. Jadi jika kedua *IF*-nya diyakini benar (pada faktor kepastiannya), maka kesimpulannya akan memiliki CF pada maksimum keduanya :

$$CF (A \text{ atau } B) = \text{Maksimum} [CF (A), CF(B)] \dots\dots\dots(2.2)$$

Pada kasus ini, CF = 85 persen untuk harga saham akan tinggi.

2.4.2 Mengkombinasikan Dua Atau Lebih Aturan.

Ada beberapa cara untuk mencapai tujuan yang sama, masing-masing dengan CF berbeda untuk sekelompok fakta yang diberikan. Pada saat dimiliki sistem berbasis pengetahuan dengan beberapa aturan terinterrelasi, masing-masing darinya menghasilkan kesimpulan yang sama tetapi faktor kepastiannya berbeda, maka setiap aturan dapat ditampilkan sebagai potongan yang mendukung kesimpulan bersama. Untuk menghitung faktor kepastian dari kesimpulan tersebut, perlu untuk mengkombinasikan bukti sebagai berikut :

Diasumsikan bahwa ada dua aturan :

R1 : *IF* laju inflasi kurang dari 5 persen
THEN harga pasar saham naik (CF = 0,7)

R2 : *IF* tingkat pengangguran kurang dari 7 persen
THEN harga pasar saham naik (CF = 0,6)

Efek kombinasinya dihitung sebagai berikut :

$$CF(R1,R2) = CF(R1) + [CF(R2)] \times [1-CF(R1)]$$

$$= CF(R1) + CF(R2) - [CF(R1)] \times [CF(R2)] \dots\dots\dots(2.3)$$

Diberikan CF(R1) = 0,7 AND CF (R2) = 0,6

$$CF(R1,R2) = 0,7 + 0,6 - (0,7)(0,6) = 0,88$$

Artinya ES memberitahukan bahwa ada 88 persen kemungkinan harga pasar saham naik.

Untuk aturan ketiga yang ditambahkan, persamaan berikut dapat digunakan :

$$CF(R1,R2,R3) = CF(R1,R2) + [CF(R3)] \times [1-CF(R1,R2)]$$

$$= CF(R1,R2) + CF(R3) - [CF(R1,R2)] \times [CF(R3)] \dots\dots(2.4)$$

Anggap bahwa aturan ketiga ditambahkan :

R3 : *IF* harga Obligasi meningkat,
THEN harga saham naik (CF = 0,85)

Sekarang anggap semua aturan benar dalam bagian IF-nya, peluang harga saham akan naik adalah :

$$CF(R1,R2,R3) = 0,88 + 0,85 - (0,88)(0,85) = 0,982$$

Artinya ada 98,2 persen peluang harga saham akan naik.

2.5 Tahapan Pengembangan Sistem Pakar

Secara garis besar tahapan pengembangan sistem pakar sebagai berikut:

1. Mengidentifikasi masalah dan kebutuhan. Mengkaji situasi dan memutuskan dengan pasti tentang masalah yang akan dikomputerisasi dan apakah dengan sistem pakar bisa lebih membantu atau tidak.
2. Menentukan masalah yang cocok. Domain masalah tidak terlalu luas.
3. Mempertimbangkan alternatif. Dalam hal ini ada 2 alternatif yaitu menggunakan sistem pakar atau komputer tradisional.
4. Menghitung pengembalian investasi. Termasuk diantaranya: biaya pembuatan sistem pakar, biaya pemeliharaan, dan biaya training.
5. Memilih alat pengembangan. Bisa digunakan software pembuat sistem pakar atau dirancang dengan bahasa pemrograman sendiri.
6. Rekayasa Pengetahuan. Perlu dilakukan penyempurnaan terhadap aturan-aturan yang sesuai.
7. Merancang sistem.
8. Melengkapi pengembangan. Termasuk pengembangan prototype apabila sistem yang telah ada sudah sesuai dengan keinginan.

9. Menguji dan mencari kesalahan sistem.
10. Memelihara sistem.

2.6 Basis Data

Basis Data terdiri atas dua kata, yaitu basis dan data. Basis dapat diartikan sebagai tempat. Sedangkan data adalah representasi fakta dunia nyata yang mewakili suatu objek seperti manusia, barang, konsep, keadaan, dan sebagainya, yang direkam dalam bentuk angka, huruf, simbol, teks, atau kombinasinya. Basis Data ialah himpunan kelompok data (arsip) yang saling berhubungan yang diorganisasi sedemikian rupa agar kelak dapat dimanfaatkan kembali dengan cepat dan mudah.

2.6.1 Model Entity Relasional

Model entity relasional dibentuk oleh dua komponen utama, yaitu entitas dan relasi.

2.6.1.1 Entitas

Entitas merupakan individu mewakili sesuatu yang nyata dan dapat dibedakan dari sesuatu yang lain. Entitas menunjuk pada individu suatu objek. Sekelompok entitas yang sejenis dan berada dalam lingkup yang sama membentuk sebuah entitas.

2.6.1.2 Relasi

Relasi menunjukkan adanya hubungan di antara sejumlah entitas yang berasal dari himpunan entitas yang berbeda. Kumpulan semua relasi di antara entitas-entitas yang terdapat pada himpunan entitas-himpunan entitas tersebut membentuk himpunan relasi. Kardinalitas relasi menunjukkan jumlah maksimum entitas yang dapat berelasi

dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi yang terjadi antara dua himpunan entitas (misalnya A dan B) dapat berupa :

1) Satu ke satu (*One to one*)

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B, dan begitu juga sebaliknya setiap entitas pada himpunan B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

2) Satu ke banyak (*One to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, tetapi tidak sebaliknya, dimana setiap entitas pada himpunan entitas B berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas A.

3) Banyak ke satu (*Many to one*)

Setiap entitas pada himpunan entitas A berhubungan dengan paling banyak dengan satu entitas pada himpunan entitas B.

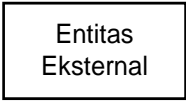
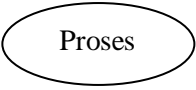
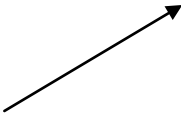

4) Banyak ke banyak (*Many to many*)

Setiap entitas pada himpunan entitas A dapat berhubungan dengan banyak entitas pada himpunan entitas B, dan juga sebaliknya, dimana setiap entitas pada himpunan entitas B dapat berhubungan dengan banyak entitas pada himpunan entitas A.

2.7 Diagram Aliran Data

Diagram aliran data/*data flow control* (DFD) adalah sebuah teknik grafis yang menggambarkan aliran informasi dan transformasi yang diaplikasikan pada saat data bergerak dari input menjadi output. DFD dapat digunakan untuk menyajikan sebuah sistem atau perangkat lunak pada setiap tingkat abstraksi.

Notasi-notasi Diagram aliran data :

1.  Prosedur atau konsumen informasi yang ada di luar Bound sistem untuk dimodelkan.
2.  Transfer informasi (fungsi) yang ada di dalam bound sistem untuk di dalam bound sistem untuk dimodelkan
3.  Objek data, anak panah menunjukkan aliran data
4.  Penyimpanan data, informasi tersimpan yang digunakan oleh perangkat lunak.

2.8 PHP

PHP dikenal sebagai sebuah bahasa scripting yang menyatu dengan tag-tag HTML, dieksekusi di server, dan digunakan untuk membuat halaman web yang dinamis seperti halnya *Active Server Pages* (ASP) atau *Java Server Pages* (JSP).

Kode Program PHP menyatu dengan tag-tag HTML dalam satu file. Kode PHP diawali dengan tag `<? atau <?php` dan ditutup dengan tag `?>`. *File* yang berisi tag HTML dan kode PHP ini diberi ekstensi `.php`. Berdasarkan ekstensi ini, pada saat *file* diakses, *server* akan tahu bahwa *file* ini mengandung kode PHP. *Server* akan menerjemahkan kode ini dan menghasilkan *output* dalam bentuk tag HTML yang akan dikirim ke *browser client* yang mengakses *file* tersebut.

PHP mendukung banyak paket database baik yang komersil maupun non komersil, seperti PostgreSQL, mSQL, MySQL, Oracle, Microsoft SQL Server, dan banyak lagi.

2.9 Penyakit Ayam

Jika dievaluasi, penyakit ayam yang terdapat di Indonesia setiap tahun bertambah dan disebabkan oleh banyak hal seperti virus, bakteri, parasit, cacing, dan sebagainya. Penyakit ayam tersebut tidak dapat dipisahkan, baik dari ayam ras, ayam buras, ayam petelur, maupun ayam pedaging. Pada umumnya penyakit-penyakit tersebut ditemukan pada kelompok-kelompok ayam tersebut.

Masalah penyakit dalam usaha peningkatan produksi ternak ayam merupakan gangguan dan ancaman yang serius. Kerugian yang ditimbulkan penyakit ayam dapat berbentuk kematian, pertumbuhan terlambat atau penurunan produksi telur. Penanganan penyakit ayam harus di program secara seksama, sempurna dan terarah. Sebab program penanganan penyakit memegang peranan yang dominan dalam peningkatan produksi ternak.

Adapun penyakit ayam yang disebabkan oleh virus dan sering muncul di Indonesia adalah :

1. *Avian Influenza* (Flu Burung)

Avian Influenza (AI) atau flu burung adalah penyakit yang disebabkan oleh virus influenza tipe A. Virus influenza tipe A dapat berubah-ubah bentuk (*Drift, Shift*), dan dapat menyebabkan epidemi dan pandemi. Virus influenza tipe A terdiri dari *Hemaglutinase* (H) dan *Neuroamidase* (N), kedua huruf ini digunakan sebagai identifikasi kode subtipe flu burung yang banyak jenisnya. Ini merupakan virus yang paling berbahaya, karena bersifat zoonosis (menyerang manusia). *Avian Influenza* dapat memberikan dampak kerugian ekonomik yang cukup tinggi dalam industri perunggasan.

Gejala yang dialami pada ayam yaitu gangguan pernapasan, jengger dan pial berwarna kemerahan sampai kebiruan, penurunan produksi telur, diare, kelumpuhan, bercak darah (borok) pada kaki, dapat menyebabkan kematian, dan kematian mendadak serta angka kematian yang tinggi.

Pengobatan untuk ayam dan unggas yang terinfeksi flu burung belum ditemukan. Pengobatan dilakukan buat manusia yang terinfeksi flu burung. Dan ayam yang terinfeksi flu burung dianjurkan untuk dimusnahkan, karena dapat menyerang manusia (*zoonosis*). Dalam melakukan pemusnahan harap memberi tahu kepada Dinas Peternakan guna dilakukan pengawasan dalam pemusnahan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi, memberikan vaksin pada ayam yang sehat yaitu *VOLVAC-AI KV* dipakai secara suntikan bawah kulit (sub cutan), memberikan multivitamin dan melaksanakan *Bio Security* secara ketat.

2. *Newcastle disease* (ND)

Newcastle disease (ND) merupakan suatu penyakit pernapasan dan sistemik, yang bersifat akut dan mudah sekali menular, yang disebabkan oleh virus *Avian Paramyxovirus* dan menyerang berbagai jenis unggas, terutama ayam. *Newcastle disease* merupakan suatu penyakit yang bersifat kompleks oleh karena isolat dan strain virus yang berbeda dapat menimbulkan variasi yang besar dalam derajat keparahan dari penyakit. *Newcastle disease* dapat memberikan dampak kerugian ekonomik yang cukup tinggi dalam industri perunggasan. Gejala yang dialami pada ayam yaitu gangguan pernapasan, diare berwarna hijau, kelemahan, kehilangan nafsu makan, kehilangan nafsu minum, penurunan produksi telur, dapat menyebabkan kematian dan angka kematian yang tinggi.

Pengobatan untuk ayam dan unggas yang terkena *Newcastle disease* belum ditemukan dan dianjurkan untuk dimusnahkan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi, memberikan multivitamin, dan memberikan vaksin pada ayam yang sehat yaitu *SOTASEC* dipakai secara suntikan IM, spray, tetes mata, lewat air minum

3. *Infectious Bronchitis* (IB)

Infectious Bronchitis (IB) merupakan penyakit yang menyerang saluran pernapasan ayam yang bersifat akut dan sangat mudah menular. Penyakit ini disebabkan oleh adanya cairan dalam trakea. Dan disebabkan oleh virus yang tergolong *single stranded RNA*, famili *coronividae* dan genus *coronavirus*.

Gejala yang dialami pada ayam yaitu gangguan pernapasan, batuk, kelemahan, bulu berdiri, penurunan produksi telur, keluarnya leleran dari hidung. Pengobatan untuk ayam yang terkena penyakit *Infectious Bronchitis* belum ditemukan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi, memberikan multivitamin, dan memberikan vaksin pada ayam yang sehat yaitu AVABRON HN-63 pemakaiannya secara tetes ke mata atau lewat air minum.

4. *Infectious Laryngotracheitis* (ILT,LT)

Infectious Laryngotracheitis (ILT,LT) merupakan suatu penyakit viral pada ayam yang bersifat akut maupun ringan, yang tersifat adanya kesulitan bernapas dan adanya eksudat bercampur darah yang berasal dari trakea dan keluar melalui hidung ataupun mulut. Dan disebabkan oleh *herpesvirus* group A yang termasuk famili *Hervesviridae*, subfamili *Alphahervesvirinae*.

Gejala yang dialami pada ayam yaitu gangguan pernafasan, batuk, mata berair, keluarnya leleran yang bercampur darah dari hidung dan mulut, mengantuk, dan dapat menyebabkan kematian.

Pengobatan untuk ayam yang terkena penyakit *Infectious Laryngotracheitis* belum ditemukan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi, memberikan multivitamin, dan memberikan vaksin pada ayam yang sehat yaitu LT-IVAX dengan pemakaiannya secara tetes ke mata.

5. *Fowl Pox* (Cacar Unggas)

Fowl Pox (cacar unggas) merupakan penyakit viral pada unggas peliharaan, yang tersifat oleh adanya lesi berbentuk nodular, yang bersifat proliferaatif dan menyebar pada kulit dari bagian tubuh yang tidak ditumbuhi oleh bulu. Disebabkan oleh virus *pox* serta dan banyak dijumpai pada daerah yang terdapat insekta penghisap darah, misalnya nyamuk serta tempat berkembang biaknya seperti rawa-rawa (Tabbu, 2002).

Gejala yang dialami pada ayam yaitu gangguan pernapasan, munculnya lesi pada daerah yang tidak di tumbuhi bulu, pembengkakan dari sinus dan mata.

Pengobatan untuk ayam yang terkena penyakit *fowl pox* belum ditemukan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi, memberikan multivitamin, dan memberikan vaksin pada ayam yang sehat yaitu AVAPOX pemakaian dengan cara tusuk kulit

6. *Infectious Bursal Disease* (IBD)

Infectious Bursal Disease (IBD) atau *Gumboro* merupakan penyakit yang bersifat akut dan sangat mudah menular. Penyakit ini merusak berbagai organ *limfoid*, sehingga ayam yang terserang akan lebih peka terhadap berbagai penyakit. Disebabkan oleh virus yang tergolong genus *Birnavirus* dan famili *Birnaviridae*.

Gejala yang dialami pada ayam yaitu kehilangan nafsu minum, warna bulu kusam, bulu berdiri, kelemahan, diare dan dapat menyebabkan kematian. Pengobatan untuk ayam yang terkena penyakit *Infectious Bursal Disease* belum ditemukan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi memberikan vaksin pada ayam yang sehat yaitu *BURSIMUNE* atau *BIOGUMBORO* pemakaian dengan cara tusuk ke kulit, memberikan multivitamin.

7. *Swollen Head Syndrome* (SHS)

Swollen Head Syndrome (SHS) merupakan penyakit yang disebabkan oleh virus *Avian Pneumovirus* yang menyebabkan nekrosis dan pendarahan pada saluran pernapasan bagian atas serta menyebabkan kebengkakan di daerah kepala.

Gejala yang dialami pada ayam yaitu gangguan pernapasan, konjungtiva kemerahan, bengkak pada kelenjar air mata, ayam cenderung menggaruk bagian muka, produksi telur menurun. Pengobatan untuk ayam yang terkena penyakit *Infectious Bursal Disease* belum ditemukan. Sedangkan pencegahan yang dilakukan agar penyakit tidak menular yaitu melakukan sanitasi dan desinfeksi, memberikan multivitamin, memberikan vaksin pada ayam yang sehat yaitu *AVIFFA-RTI* pemakaiannya tetes mata atau air minum.