

## BAB II

### LANDASAN TEORI

#### 2.1 Sistem Pakar

Sistem pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut. Sistem pakar bekerja berdasarkan pengetahuan yang dimasukkan oleh seorang atau beberapa orang pakar dalam rangka mengumpulkan informasi hingga sistem pakar dapat menemukan jawabannya (Kusrini,2006).

Secara umum dikatakan bahwa sistem pakar adalah program komputer yang menirukan penalaran seorang pakar dengan keahlian pada suatu wilayah pengetahuan tertentu. Sistem pakar merupakan program “*artificial intelligence*” (“kecerdasan buatan” atau AI) yang menggabungkan basis pengetahuan dengan mesin inferensi. Hal ini merupakan bagian perangkat lunak spesialisasi tingkat tinggi atau bahasa pemrograman tingkat tinggi (*High Level Language*), yang berusaha menduplikasi fungsi seorang pakar dalam satu bidang keahlian tertentu. Program ini bertindak sebagai konsultan yang cerdas atau penasihat dalam suatu lingkungan keahlian tertentu, sebagai hasil himpunan pengetahuan yang telah dikumpulkan dari beberapa orang pakar. Dengan demikian seorang awam sekalipun bisa menggunakan sistem pakar itu untuk memecahkan berbagai persoalan yang ia hadapi dan bagi seorang ahli, sistem pakar dapat dijadikan alat untuk menunjang aktivitasnya yaitu sebagai sebagai asisten yang berpengalaman. Sistem pakar yang muncul pertama kali adalah *General-purpose problem solver* (GPS) yang dikembangkan oleh Newl dan Simon (Septian, 2013)

### 2.1.1 Struktur Sistem Pakar

Menurut Septian (2013) sistem pakar memiliki dua bagian utama, yaitu :

- a) Lingkungan pengembangan (*development environment*), yaitu bagian yang digunakan untuk memasukkan pengetahuan pakar ke dalam lingkungan sistem pakar.
- b) Lingkungan konsultasi (*consultation environment*), yaitu bagian yang digunakan oleh pengguna yang bukan pakar untuk memperoleh pengetahuan.

Sistem pakar memiliki 3 bagian, yaitu antar muka pengguna, mesin inferensi dan basis pengetahuan.

#### 1. Antar Muka Pengguna

Antar muka pengguna adalah perangkat lunak yang menyediakan media komunikasi antara pengguna dengan sistem. Antar muka pengguna memberikan fasilitas informasi dan berbagai keterangan yang bertujuan untuk membantu mengarahkan alur penelusuran masalah sehingga ditemukan sebuah solusi.

#### 2. Mesin Inferensi

Mesin Inferensi (*Inference Engine*), merupakan otak dari Sistem Pakar, juga dikenal sebagai penerjemah aturan (*rule interpreter*). Komponen ini mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu masalah. Mesin inferensi adalah program komputer yang memberikan metodologi untuk penalaran tentang informasi yang ada dalam basis pengetahuan dan dalam *workplace*, dan untuk memformulasikan kesimpulan.

Kerja mesin inferensi meliputi:

- Menentukan aturan mana akan dipakai
- Menyajikan pertanyaan kepada pemakai, ketika diperlukan.
- Menambahkan jawaban ke dalam memori Sistem Pakar.

- Menyimpulkan fakta baru dari sebuah aturan.
- Menambahkan fakta tadi ke dalam memori.

### **3. Basis Pengetahuan**

Basis pengetahuan merupakan inti program sistem pakar. Pengetahuan ini merupakan representasi pengetahuan dari seorang pakar. Basis pengetahuan ini bisa direpresentasikan dalam berbagai macam bentuk, salah satunya adalah bentuk *case based reasoning* yang ditemukan dari ide untuk menggunakan pengalaman yang terdokumentasi dalam menyelesaikan masalah yang baru. Ada 2 bentuk pendekatan basis pengetahuan (Kusumadewi, 2003) yaitu penalaran berbasis aturan (*rule-based reasoning*) dan penalaran berbasis kasus (*case based reasoning*).

Akuisisi pengetahuan adalah akumulasi , transfer, dan transformasi keahlian dalam menyelesaikan masalah dari sumber pengetahuan ke dalam program komputer.

#### **2.1.2 Keuntungan dan Kelemahan Sistem Pakar**

##### **Keuntungan sistem pakar:**

- a) Memungkinkan orang awam dapat mengerjakan pekerjaan para ahli.
- b) Dapat melakukan proses secara berulang secara otomatis.
- c) Menyimpan pengetahuan dan keahlian para pakar.
- d) Meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan.
- e) Meningkatkan hasil dan produktivitas, karena sistem pakar dapat bekerja lebih cepat dari manusia.
- f) Memiliki kemampuan untuk bekerja dengan informasi yang tidak lengkap dan mengandung ketidakpastian.
- g) Memiliki kemampuan untuk mengakses pengetahuan.
- h) Meningkatkan kapabilitas dalam penyelesaian masalah.
- i) Menghemat waktu dalam pengambilan keputusan.

### **Kelemahan sistem pakar:**

- a) Biaya yang diperlukan untuk membuat, memelihara, dan mengembangkannya sangat mahal.
- b) Sulit dikembangkan, hal ini erat kaitannya dengan ketersediaan pakar di bidangnya.
- c) Sistem pakar tidak 100% benar karena seseorang yang terlibat dalam pembuatan sistem pakar tidak selalu benar. Oleh karena itu perlu diuji ulang secara teliti sebelum digunakan.
- d) Kepakaran sangat sulit diekstrak dari manusia.
- e) Pendekatan oleh setiap pakar untuk suatu situasi atau problem bisa berbeda-beda, meskipun sama-sama benar.
- f) Sangat sulit bagi seorang pakar untuk mengabstraksi atau menjelaskan langkah mereka dalam menangani masalah.
- g) Sistem pakar bekerja baik untuk suatu bidang yang sempit.
- h) Istilah yang dipakai oleh pakar dalam mengekspresikan fakta seringkali terbatas dan tidak mudah dimengerti oleh orang lain.
- i) Transfer pengetahuan dapat bersifat subyektif dan biasa.

## **2.2 Metode *Case-Based Reasoning* (CBR)**

*Case-Based Reasoning* (CBR) adalah metode untuk menyelesaikan masalah dengan mengingat kejadian-kejadian yang sama/sejenis (*similar*) yang pernah terjadi di masa lalu kemudian menggunakan pengetahuan/informasi tersebut untuk menyelesaikan masalah yang baru, atau dengan kata lain menyelesaikan masalah dengan mengadaptasi solusi-solusi yang pernah digunakan di masa lalu (Octaviani, 2008)

Menurut Aamodt dan Plaza (1994) *Case-Based Reasoning* adalah suatu pendekatan untuk menyelesaikan suatu permasalahan (*problem solving*) berdasarkan solusi dari permasalahan sebelumnya. Suatu masalah baru dipecahkan dengan menemukan kasus yang serupa di masa lampau, dan

menggunakannya kembali pada situasi masalah yang baru. Pendekatan lain dari CBR yang tidak kalah penting adalah CBR juga merupakan suatu pendekatan ke arah *incremental* yaitu pembelajaran terus-menerus.

Dalam *Case-Based Reasoning* ada empat tahapan yang meliputi :

**a. Retrieve**

Mendapatkan/memperoleh kembali kasus yang paling menyerupai/relevan (*similar*) dengan kasus yang baru. Tahap *retrieval* ini dimulai dengan menggambarkan/menguraikan sebagian masalah, dan diakhiri jika ditemukannya kecocokan terhadap masalah sebelumnya yang tingkat kecocokannya paling tinggi. Bagian ini mengacu pada segi identifikasi, kecocokan awal, pencarian dan pemilihan serta eksekusi.

**b. Reuse**

Memodelkan/menggunakan kembali pengetahuan dan informasi kasus lama berdasarkan bobot kemiripan yang paling relevan ke dalam kasus yang baru, sehingga menghasilkan usulan solusi dimana mungkin diperlukan suatu adaptasi dengan masalah yang baru tersebut.

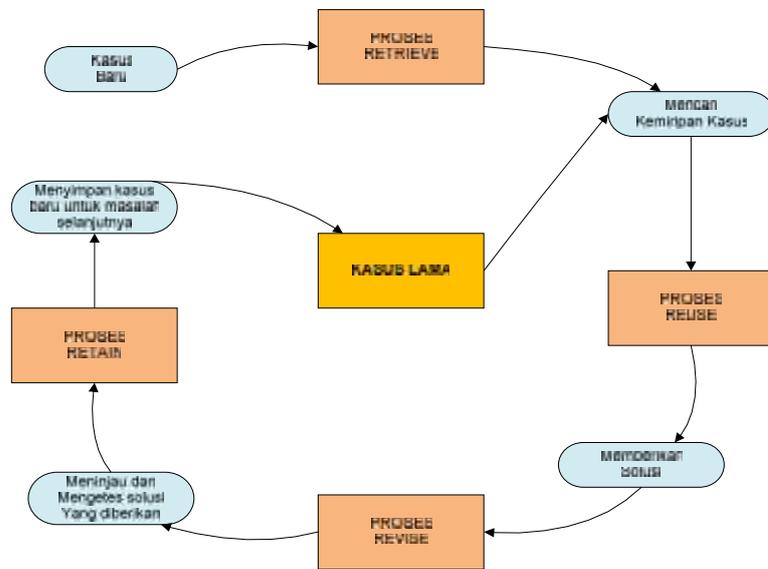
**c. Revise**

Meninjau kembali solusi yang diusulkan kemudian mengetesnya pada kasus nyata (simulasi) dan jika diperlukan memperbaiki solusi tersebut agar cocok dengan kasus yang baru.

**d. Retain**

Mengintegrasikan/menyimpan kasus baru yang telah berhasil mendapatkan solusi agar dapat digunakan oleh kasus-kasus selanjutnya yang mirip dengan kasus tersebut. Tetapi jika solusi baru tersebut gagal, maka menjelaskan kegagalannya, memperbaiki solusi yang digunakan, dan mengujinya lagi.

Empat proses masing-masing melibatkan sejumlah langkah-langkah spesifik, yang akan dijelaskan pada Gambar 2.1 berikut ini :



**Gambar 2.1. Siklus Metode *Case-Based Reasoning***

Pada saat terjadi permasalahan baru, pertama-tama sistem akan melakukan proses *retrieve*. Proses *retrieve* akan melakukan dua langkah pemrosesan, yaitu pengenalan masalah dan pencarian persamaan masalah pada database.

Setelah proses *retrieve* selesai dilakukan, selanjutnya system akan melakukan proses *reuse*. Di dalam proses *reuse*, sistem akan menggunakan informasi permasalahan sebelumnya yang memiliki kesamaan untuk menyelesaikan permasalahan yang baru. Pada proses *reuse* akan menyalin, menyeleksi, dan melengkapi informasi yang akan digunakan.

Selanjutnya pada proses *revise*, informasi tersebut akan dikalkulasi, dievaluasi, dan diperbaiki kembali untuk mengatasi kesalahan-kesalahan yang terjadi pada permasalahan yang baru.

Pada proses terakhir, system akan melakukan proses *retain*. Proses *retain* akan mengindeks, mengintegrasikan, dan mengekstrak solusi yang baru tersebut ke dalam database. Selanjutnya, solusi ke dalam basis baru itu akan disimpan ke dalam basis pengetahuan (*knowledge-base*) untuk menyelesaikan permasalahan yang akan datang. Tentunya, permasalahan yang akan diselesaikan adalah permasalahan yang memiliki kesamaan dengannya.

### 2.3 Penerapan Algoritma *Nearest Neighbor Retrieval* Pada Metode *Case-Based Reasoning*

Algoritma *Nearest Neighbor Retrieval* (*K-nearest neighbor* atau *K-NN*) adalah sebuah algoritma untuk melakukan klasifikasi terhadap objek berdasarkan data pembelajaran yang jaraknya paling dekat dengan objek tersebut. Kasus khusus di mana klasifikasi diprediksikan berdasarkan data pembelajaran yang paling dekat (dengan kata lain,  $k = 1$ ) disebut algoritma *nearest neighbor*.

Algoritma *nearest neighbor* berdasarkan pada proses pembelajaran menggunakan analogi / *learning by analogi*. *Training* sampelnya dideskripsikan dalam bentuk atribut numerik  $n$ -dimensi. Dengan cara ini, semua *training* sampel disimpan pada pola ruang  $n$ -dimensi. Ketika diberikan "*unknown*" sampel, *k-nearest neighbor classifier* mencari pola ruang  $K$  *training* sampel yang paling dekat "*unknown*" sampel tersebut.  $K$  *training* sampel ini adalah  $k$  *nearest neighbor* dari *unknown* sampel. *Unknown* sampel ditetapkan dengan class yang paling umum diantara  $k$  *nearest neighbors*nya. Ketika  $k = 1$ , *unknown* sampel ditetapkan dengan class dari *training* sampel yang paling dekat dengan pola ruangnya.

Algoritma *nearest neighbor retrieval* menyimpan semua *training* sampel dan tidak membangun *classifier* sampai sampel baru (*unlabeled*) perlu diklasifikasikan, sehingga algoritma *nearest neighbor retrieval* sering disebut dengan *instance-based* atau *lazy learners*.

Rumus untuk menghitung bobot kemiripan (*similarity*) dengan *nearest neighbor retrieval* adalah (Octaviani, 2008):

$$\text{Similarity (problem, case)} = \frac{S_1 \cdot W_1 + S_2 \cdot W_2 + \dots + S_n \cdot W_n}{W_1 + W_2 + \dots + W_n} \quad (2.1)$$

Keterangan:

$S$  = *similarity* (nilai kemiripan) yaitu 1 (sama) dan 0 (beda)

$W$  = *weight* (bobot yang diberikan)

<b>Bobot Parameter (w)</b>	
Gejala Penting	5
Gejala Sedang	3
Gejala Biasa	1

## 2.4 Penyakit Paru

Paru-paru merupakan suatu organ tubuh yang berfungsi sebagai alat pernafasan manusia. Didalam paru-paru ini terjadi proses pertukaran gas oksigen (O<sub>2</sub>) dan karbondioksida (CO<sub>2</sub>). Setelah membebaskan oksigen, sel-sel darah merah menangkap karbondioksida sebagai hasil metabolisme tubuh yang akan dibawa ke paru-paru. Di paru-paru karbondioksida dan uap air dilepaskan dan dikeluarkan dari paru-paru melalui hidung.

Paru-paru merupakan salah satu organ yang sangat penting bagi manusia. Baik karena fungsinya maupun sebagai salah satu organ yang rentan terhadap berbagai penyakit. Penyakit pada paru-paru bisa disebabkan karena bawaan dari lahir dan bisa juga karena faktor atau kebiasaan buruk yang sering dilakukan dalam kehidupan sehari-hari. Kebiasaan buruk yang bisa menyebabkan penyakit pada paru yaitu : Merokok, minum-minuman beralkohol, menghirup gas karbon terlalu berlebihan, lingkungan hidup yang tidak bersih, dsb.

Adapun beberapa penyakit paru, yaitu :

### 1. Radang Paru-Paru ( Pneumonia)

Pneumonia dapat timbul di berbagai daerah di paru-paru. Pneumonia lobar menyerang sebuah lobus atau potongan besar paru-paru. Penyebab utama penyakit ini merupakan infeksi bakteri, sering kali dari jenis *Streptococcus Pneumoniae*. Bakteri ini menjadi permasalahan sekunder oleh infeksi virus di saluran pernapasan atas, seperti flu.

Gejala umum penderita penyakit ini adalah :

- Badan lemah

- Batuk berdahak kuning
- Berat badan turun
- Demam lebih dari seminggu
- Keringat malam
- Nyeri di dada
- Nafsu makan turun
- Panas naik turun
- Sakit kepala
- Sesak nafas

## **2. Legionnaires**

Penyakit ini disebabkan oleh bakteri legionella yang merupakan bakteri berbentuk batang yang ditemukan di sebagian besar sumber air. Bakteri ini berlipat ganda dengan sangat cepat. Bakteri ini sering berasarang di pipa ledeng atau dimanapun air tergenang. Bagi manusia yang memiliki system kekebalan tubuh yang lemah, penyakit ini bisa mengakibatkan kematian.

Gejala umum penderita penyakit ini adalah :

- Sesak napas
- Batuk berdahak putih
- Demam lebih dari seminggu
- Mual
- Muntah
- Panas naik turun
- Perut terasa sakit

## **3. Tuberkulosis (TB)**

Penyebab penyakit ini disebabkan oleh bakteri mycobacterium tuberculosis. Sebagian besar manusia memiliki mikroba TB di dalam

tubuhnya, namun mikroba ini hanya menyebabkan penyakit pada seseorang yang memiliki daya daya imun yang rendah.

Gejala umum penderita penyakit ini adalah :

- Badan lemah
- Batuk berdarah
- Sesak napas
- Nafsu makan menurun
- Berat badan turun
- Muntah
- Panas naik turun

#### **4. Sesak nafas (asma)**

Asma adalah penyakit paru-paru yang menimbulkan serangan sesak napas dan mengi yang berulang. Asma merupakan salah satu kelainan paru-paru paling banyak dan bervariasi. Otot dinding saluran udara berkontraksi seperti kejang, menyebabkan saluran udara menyempit, sehingga terjadi serangan sesak napas.

Gejala umum penderita penyakit ini adalah :

- Sesak napas
- Batuk berdahak putih
- Batuk hilang timbul
- Bersin-bersin di pagi hari
- Nyeri di dada
- Riwayat asma dalam keluarga positif
- Sesak nafas dipicu udara dingin

#### **5. Bronkitis**

Penyakit ini disebabkan oleh rokok. Pada bronkitis, bronkus, saluran utama menuju paru-paru meradang, membengkak, dan menyempit akibat iritasi oleh asap tembakau, infeksi berulang. Saluran

udara yang meradang menyebabkan dahak yang berlebihan , awalnya menyebabkan batuk mengganggu di waktu lembab dan dingin, lalu berlanjut sepanjang tahun.

Gejala umum penderita penyakit ini adalah :

- Alergi debu
- Sesak napas
- Keringat malam
- Batuk berdahak putih
- Panas naik turun
- Suara serak

## **6. Emfisema**

Emfisema adalah jenis penyakit paru obstruktif kronik yang melibatkan kerusakan pada kantung udara (alveoli) di paru-paru. Akibatnya, tubuh tidak mendapatkan oksigen yang diperlukan. Penyebab paling utama adalah perokok berat.

Gejala umum penderita penyakit ini adalah :

- Sesak napas
- Batuk berdahak putih
- Berat badan turun
- Demam lebih dari seminggu
- Perokok aktif
- Sakit kepala

## **7. Kanker paru-paru**

Keberadaan tumor ganas pada paru-paru disebut kanker paru-paru. Kanker paru-paru adalah kanker paling umum di dunia dan lebih dari satu juta kasus baru ditemukan setiap tahun. Penyebab kanker paru-paru adalah rokok, minum-minuman beralkohol, dan kurangnya berolahraga

Gejala umum penderita penyakit ini adalah :

- Batuk berdarah
- Batuk hilang timbul
- Berat badan turun
- Demam lebih dari seminggu
- Keringat malam
- Nafsu makan turun
- Nyeri di dada
- Nyeri di punggung
- Panas naik turun
- Perut terasa sakit
- Riwayat kanker dalam keluarga positif
- Suara serak
- Sulit menelan
- Sesak nafas

## 2.5 PHP

Pertama kali PHP dibuat oleh Rasmus Lerdorf pada tahun 1995 menggunakan nama PHP/FI yang memiliki kepanjangan *Personal Home Page/Form Interpreter*. Namun pada perkembangannya, pada tahun 1997, Andi Gutmans dan Zeev Suraski menulis ulang PHP yang kemudian sampai sekarang dikenal kependekan dari kata *Hypertext Preprocessor*. PHP tergolong sebagai perangkat lunak *open source* yang diatur dalam aturan *general purpose licences* (GPL). Pemrograman PHP sangat cocok dikembangkan dalam lingkungan *web*, karena PHP dikhususkan untuk pengembangan *web* dinamis. Maksudnya PHP mampu menghasilkan *website* yang terus menerus hasilnya bisa berubah-ubah sesuai dengan pola yang diberikan (Septian, 2013).

## 2.6 MySQL

MySQL adalah suatu sistem manajemen *database*. Suatu *database* adalah suatu koleksi data terstruktur. Data tersebut dapat berupa apa saja, dari *list* sederhana sampai sebuah galeri gambar. Untuk menambah, mengakses, dan memproses data yang tersimpan dalam sebuah database, dibutuhkan suatu sistem manajemen database seperti halnya MySQL. Sejak komputer menjadi suatu alat yang digunakan untuk menanggulangi data dalam ukuran besar, manajemen *database* memegang peranan utama dalam perhitungan, sebagai utilitas tunggal maupun sebagai bagian dari aplikasi lain (Septian, 2013).