

## **BAB II**

### **LANDASAN TEORI**

#### **2.1. Sistem Pakar**

Sistem Pakar adalah sistem berbasis komputer yang menggunakan pengetahuan, fakta dan teknik aktivasi cerdas untuk mengatasi masalah yang biasanya dapat diatasi oleh seorang pakar pada bidang tertentu ( Kusrini 2006 ). Sistem pakar juga dapat berfungsi sebagai asisten yang pandai dari seorang pakar. Alasan mendasar mengapa sistem pakar dikembangkan untuk menggantikan seorang pakar adalah :

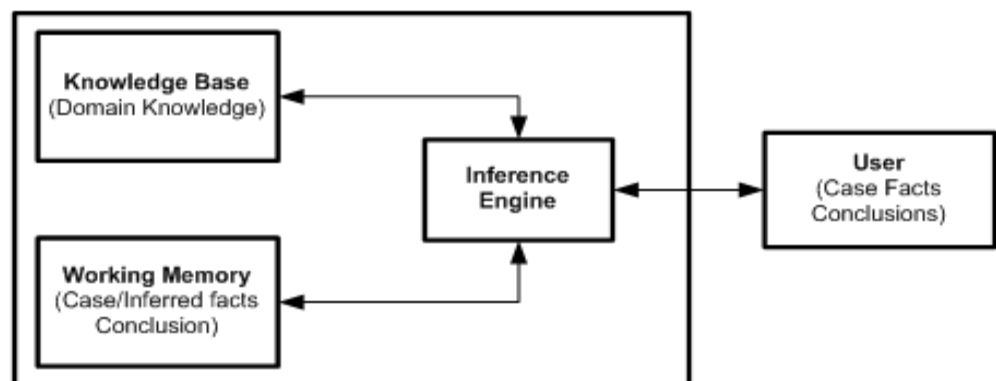
1. Dapat menyediakan kepakaran setiap waktu dan di berbagai lokasi .
2. Secara otomatis mengerjakan tugas-tugas rutin yang membutuhkan seorang pakar.
3. Seorang pakar akan pensiun atau pergi.
4. Menghadirkan atau menggunakan jasa seorang pakar memerlukan biaya yang mahal.
5. Kepakaran dibutuhkan juga pada lingkungan yang tidak bersahabat.

Keuntungan pemakaian sistem pakar (Kusrini, 2006) yaitu :

1. Menyimpan pengetahuan dan keahlian para pakar.
2. Dapat bekerja dengan informasi yang tidak lengkap atau tidak pasti.
3. Meningkatkan output dan produktifitas.
4. Meningkatkan kualitas.

5. Menyediakan nasihat yang konsisten dan dapat mengurangi tingkat kesalahan.
6. Membuat peralatan yang kompleks lebih mudah dioperasikan karena sistem pakar dapat melatih pekerja yang tidak berpengalaman.
7. Handal (reliability).
8. Sistem pakar tidak dapat lelah atau bosan, konsisten dalam memberi jawaban dan selalu memberikan perhatian penuh.
9. Memiliki kemampuan dalam memecahkan suatu permasalahan yang kompleks.
10. Memungkinkan pemindahan pengetahuan ke lokasi yang jauh serta memperluas jangkauan pakar, dapat diperoleh dan dipakai dimana saja.

Menurut Irawan (2007), sistem pakar terdiri dari tiga komponen utama, yaitu knowledge base, working memory, dan inference engine, seperti pada gambar 2.1.



**Gambar 2.1. Komponen Sistem Pakar**

### 1. Knowledge Base

Knowledge base adalah bagian dari sistem pakar yang mengandung / menyimpan pengetahuan (representasi pengetahuan dari seorang pakar).

### 2. Working Memory

*Working memory* menyimpan fakta-fakta (baik yang dimasukkan oleh pengguna maupun fakta baru) yang ditemukan selama proses konsultasi dengan sistem pakar.

### 3. Inference Engine

Inference engine adalah bagian dari sistem pakar yang bertugas mencari padanan antara fakta yang ada di dalam working memory dengan fakta-fakta tentang pengetahuan tertentu dari pakar yang ada di dalam knowledge base, kemudian menarik kesimpulan dari masalah yang diajukan kepada sistem.

## 2.2. Metode Naive Bayes

Merupakan pengklasifikasian dengan metode probabilitas dan statistik yang dilakukan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang dimasa depan berdasarkan pengalamana dimasa sebelumnya sehingga dikenal sebagai Thomas Bayes. Naïve bayes untuk setiap kelas keputusan, menghitung probabilitas dengan syarat bahwa kelas keputusan adalah benar, mengingat vektor informasi obyek. Algoritma ini mengasumsikan bahwa atribut obyek adalah independen. Probabilitas yang terlibat data memproses produksi perkiraan akhir dihitung

sebagai jumlah frekuensi dari “master” table keputusan . (Olson dan Delen,2008)

Pengklasifikasian Bayesian memiliki tingkat kesalahan minimal dibandingkan dengan klasifikasi lainnya. Namun, dalam prakteknya hal ini tidak selalu terjadi, karena ketidakakuratan asumsi yang dibuat untuk penggunaannya, seperti kondisi kelas independen, dan kurangnya data probabilitas yang tersedia. Pengklasifikasian Bayesian juga berguna dalam memberikan pembenaran teoritis untuk pengklasifikasian lain yang tidak secara eksplisit menggunakan teorema Bayes.

Menurut Han dan Kamber (2011) Proses dari The Naïve Bayesian Classifier atau Simple Bayesian Classifier, sebagai berikut :

1. Variable D menjadi pelatihan set tuple dan label yang terkait dengan kelas. Seperti biasa, setiap tuple diwakili oleh vektor atribut n-dimensi,  $X = (x_1, x_2, \dots, x_n)$ , ini menggambarkan pengukuran n dibuat pada tuple dari atribut n, masing-masing,  $A_1, A_2, \dots, A_n$ .
2. Misalkan ada kelas m,  $C_1, C_2, \dots, C_m$ . Diberi sebuah *tuple*,  $X$ , *classifier* akan memprediksi  $X$  yang masuk kelompok memiliki probabilitas posterior tertinggi, kondisi-disebutkan pada  $X$ . Artinya, *classifier naive bayesian* memprediksi bahwa  $X$  *tuple* milik kelas  $C_i$  jika dan hanya jika :

$$P(C_i|X) > P(C_j|X) \quad \text{for } 1 \leq j \leq m, j \neq i.$$

(2.1)

Jadi memaksimalkan  $P(C_i | X)$ .  $C_i$  kelas yang  $P(C_i | X)$

dimaksimalkan disebut hipotesis posteriori maksimal. Dengan teorema Bayes :

$$P(C_i|X) = \frac{P(X|C_i)P(C_i)}{P(X)}$$

(2.2)

Keterangan :

$P(C_i|X)$  = Probabilitas hipotesis  $C_i$  jika diberikan fakta atau *record*  $X$  (*Posterior probability*)

$P(X|C_i)$  = mencari nilai parameter yang memberi kemungkinan yang paling besar (*likelihood*)

$P(C_i)$  = Prior probability dari  $X$  (*Prior*

*probability*)  $P(X)$  = Jumlah probability

*tuple* yg muncul

3. Ketika  $P(X)$  adalah konstan untuk semua kelas, hanya  $P(X|C_i)P(C_i)$  butuh dimaksimalkan. Jika probabilitas kelas sebelumnya tidak diketahui, maka umumnya diasumsikan ke dalam kelas yang sama, yaitu,  $P(C_1) = P(C_2) = \dots = P(C_m)$ , maka dari itu akan memaksimalkan  $P(X|C_i)$ . Jika tidak, maka akan memaksimalkan  $P(X|C_i)P(C_i)$ . Perhatikan bahwa probabilitas sebelum kelas dapat diperkirakan oleh  $P(C_i) = \frac{|C_i|}{|D|}$ , dimana  $|C_i|$  adalah jumlah *tuple* pelatihan kelas  $C_i$  di  $D$ .

4. Mengingat *dataset* mempunyai banyak atribut, maka akan sangat sulit dalam mengkomputasi untuk menghitung  $P(X|C_i)$ . Agar dapat mengurangi perhitungan dalam mengevaluasi  $P(X|C_i)$ , asumsi *naïve* independensi kelas bersyarat dibuat. Dianggap bahwa nilai-nilai dari atribut adalah kondisional independen satu sama lain, diberikan kelas label dari *tuple* (yaitu bahwa tidak ada hubungan ketergantungan diantara atribut) dengan demikian :

$$\begin{aligned} P(X|C_i) &= \prod_{k=1}^n P(x_k|C_i) \\ &= P(x_1|C_i) \times P(x_2|C_i) \times \dots \times P(x_n|C_i). \end{aligned} \quad (2.3)$$

Maka dapat dengan mudah memperkirakan probabilitas  $P(x_1 | C_i)$ ,  $P(x_2 | C_i)$ , . . . ,  $P(x_n | C_i)$  dari pelatihan *tuple*. Ingat bahwa di sini  $x_k$  mengacu pada nilai atribut  $A_k$  untuk *tuple*  $X$ . Untuk setiap atribut, dilihat dari apakah atribut tersebut kategorikal atau *continuous-valued* Misalnya, untuk menghitung  $P(X | C_i)$  mempertimbangkan hal-hal berikut:

- a) Jika  $A_k$  adalah kategorikal, maka  $P(X_k | C_i)$  adalah jumlah *tuple* kelas  $C_i$  di  $D$  memiliki nilai  $X_k$  untuk atribut  $A_k$ , dibagi dengan  $|C_i, D|$ , jumlah *tuple* kelas  $C_i$  di  $D$ .
- b) Jika  $A_k$  *continuous-valued* , maka perlu melakukan sedikit lebih banyak pekerjaan, tapi perhitungannya cukup sederhana. Sebuah atribut *continuous-valued* biasanya diasumsikan memiliki distribusi *Gaussian* dengan rata-rata

$\mu$  dan standar deviasi  $\sigma$ , didefinisikan oleh

$$g(x, \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}}, \quad (2.4)$$

sehingga :

$$P(x_k | C_i) = g(x_k, \mu_{C_i}, \sigma_{C_i}). \quad (2.5)$$

Setelah itu hitung  $\mu_{C_i}$  dan  $\sigma_{C_i}$ , yang merupakan deviasi *mean* (rata-rata) dan standar masing-masing nilai atribut  $A_k$  untuk *tuple* pelatihan kelas  $C_i$ . Setelah itu gunakan kedua kuantitas dalam Persamaan, bersama-sama dengan  $x_k$ , untuk memperkirakan

$P(x_k | C_i)$ .

5. Untuk memprediksi label kelas  $x$ ,  $P(X|C_i)P(C_i)$  dievaluasi untuk setiap kelas  $C_i$ . *Classifier* memprediksi kelas label dari *tuple*  $x$  adalah kelas  $C_i$ , jika

$$P(X|C_i)P(C_i) > P(X|C_j)P(C_j) \quad \text{for } 1 \leq j \leq m, j \neq i.$$

(2.6)

Dengan kata lain, label kelas diprediksi adalah  $C_i$  yang mana  $P(X | C_i) P(C_i)$  adalah maksimal.

Beberapa penerapan algoritma Naive Bayes beserta masalah dan hasilnya yang pernah dilakukan diambil dari jurnal yaitu

1. Nia,2013, APLIKASI DIAGNOSA KANKER

KANDUNGAN DENGAN MENGGUNAKAN METODE

NAIVE BAYES. Penyakit ini menyerang baik itu anak-

anak,wanita maupun pria. Salah satu penyakit kanker yang

sering menyerang wanita adalah kanker kadungan. Hal ini

diakibatkan kurangnya kepedulian terhadap kesehatan

reproduksi wanita dan masih sedikitnya fasilitas untuk

mendeteksi dini penyakit kanker kandungan. Oleh karena itu,

perlu adanya aplikasi diagnosa penyakit kanker kandungan

(Kanker Rahim). Dalam tugas akhir ini, kita membangun

aplikasi diagnosa kanker kandungan dengan metode naive

bayes. Aplikasi ini akan mendiagnosa seseorang berdasarkan

gejala yang dimasukkan. Kemudian dari gejala yang

dimasukkan, aplikasi akan menghitung nilai kemungkinan

masing-masing gejala dengan mengacu data training yang

ada. Dari hasil perbandingan tersebut maka seseorang dapat

didiagnosa mengidap kanker kandungan atau tidak.



2. Alvino, Rachman Dwi (2014) PREDIKSI NASABAH YANG BERPOTENSI MEMBUKA SIMPANAN DEPOSITO MENGGUNAKAN NAIVE BAYES BERBASIS PRACTICLE SWARM OPTIMIZATION.

Deposito masih merupakan pilihan utama masyarakat untuk berinvestasi dan hal ini merupakan kesempatan bagi bank untuk menentukan strategi pemasaran dan promosi yang lebih efisien dan efektif. Atas dasar permasalahan tersebut, maka dilakukan penelitian untuk memprediksi nasabah yang berpotensi membuka deposito dengan menggunakan teknik data mining khususnya algoritma Naive Bayes berbasis PSO. PSO pada penelitian ini akan digunakan untuk seleksi fitur. Hasil dari penelitian ini, akurasi algoritma Naive Bayes adalah 82,19% dan akurasi algoritma Naive Bayes berbasis PSO adalah 89,70%. Penggunaan algoritma PSO ternyata meningkatkan akurasi sebesar 7,51% dan algoritma tersebut dapat digunakan untuk sistem pendukung keputusan pada penelitian ini.

### 2.3. Personal Hypertext Preprocessor (PHP)

Pada awalnya PHP merupakan singkatan dari personal Home Page tools, yang gunanya untuk memonitor pengunjung web. PHP mula-mula dikembangkan oleh Rasmus Lerdorf. Kemudian istilah PHP mengacu pada Hypertext Preprocessor. PHP merupakan bahasa berbentuk script yang ditempatkan dalam server dan diproses di server. Hasilnya akan dikirim ke client, tempat pemakai menggunakan browser. PHP dikenal sebagai 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). PHP merupakan sebuah software open source. Nah, hal inilah yang membedakan ASP dengan PHP. (Swastika, 2006).

Kelebihan PHP dari bahasa pemrograman lainnya yaitu :

- a. Bahasa pemrograman PHP adalah sebuah bahasa script yang tidak melakukan sebuah kompilasi dalam penggunaannya.
- b. Web Server yang mendukung PHP dapat ditemukan dimana-mana dari mulai apache, IIS, Lighttpd, hingga Xitami dengan konfigurasi yang relative mudah.
- c. Dalam sisi pengembangan lebih mudah, karena banyaknya milis-milis dan developer yang siao membantu dalam pengembangan.
- d. Dalam sisi pemahaman, PHP adalah bahasa scripting yang paling mudah karena memiliki referensi yang banyak.

## 2.4. MySQL

MySQL adalah merupakan perangkat lunak untuk system manajemen database (database management system). Karena sifatnya yang open source dan memiliki kemampuan menampung kapasitas yang sangat besar, maka MySQL menjadi database yang sangat populer dikalangan programmer web. (Sukarno, 2006)

Pada bulan Mei 1996, MySQL versi 1.0 berhasil dirilis namun penggunaanya terbatas 4 orang saja, namun dibulan Oktober ditahun yang sama versi 3.11.0 dilepaskan ke public tapi belum bersifat open source. Bulan Juni 2000, MySQL AB mengumumkan bahwa sejak versi 3.23.19, MySQL adalah merupakan software database yang bebas berlisensi GPL atau General Public Licensi yang open source. Mulanya MySQL hanya berjalan disystem operasi linu namun pada saat MySQL versi 3.22 tahun 1998-1999 sudah tersedia diberbagai platform termasuk windows. Ini terjadi karena MySQL menjadi semakin populer dan dilirik banyak orang karena kestabilan dan kecepatan yang meningkat. Beberapa keunggulan dari MySQL adalah :

- a. Mampu menangani jutaan user dalam waktu bersamaan
- b. Mampu menanggung lebih dari 50.000.000 record.
- c. Sangat cepat mengeksekusi perintah
- d. Memiliki user privilege yang mudah dan efisien