



Hak cipta dan penggunaan kembali:

Lisensi ini mengizinkan setiap orang untuk menggubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

Copyright and reuse:

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

BAB II

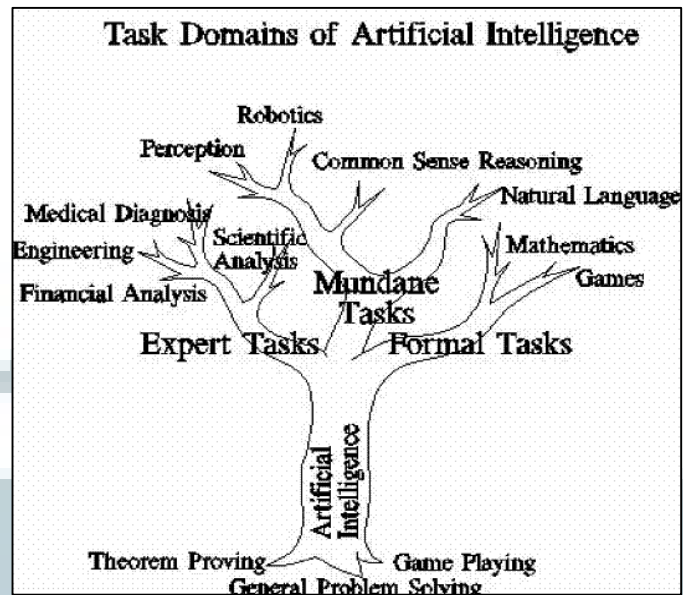
LANDASAN TEORI

2.1 Artificial Intelligence

Artificial Intelligence (AI) adalah sebuah cabang ilmu dari *computer science* yang memiliki tujuan untuk menghasilkan “kecerdasan” pada *behavior* mesin. Suatu mesin yang “cerdas” dapat berpikir secara logis dan melakukan aksi secara rasional berdasarkan pemikiran tersebut layaknya seorang manusia. Istilah AI pertama kali dikemukakan sejak tahun 1956 pada konferensi Dartmouth oleh John McCarthy (Suyanto, 2014).

Meski istilah tersebut baru dikemukakan pada tahun 1956, penelitian mengenai konsep AI sudah dimulai sejak tahun 1940-an dengan ditemukannya alat penyimpanan yang dinamakan dengan komputer elektronik. Komputer elektronik dapat menyimpan program dan memroses informasi, sehingga memudahkan pekerjaan para *programmer* pada waktu itu (Suyanto, 2014). Pada tahun-tahun berikutnya, penelitian AI berkembang menjadi tiga (3) *domain* seperti pada gambar 2.1.

Berdasarkan definisi AI yaitu *acting rationally*, AI melakukan pendekatan dengan konsep *rational agent*. *Rational agent* adalah *agent* yang bertindak dan mengamati *environment*, sehingga menghasilkan *output* sesuai dengan tujuan yang rasional (Suyanto, 2014). *Agent* memiliki tujuan yang diukur dengan *performance measure*. Sebuah *rational agent* selalu bertindak untuk memaksimalkan ukuran kinerja dengan menggunakan pengetahuan yang dimilikinya.



Gambar 2.1 *Task Domains of Artificial Intelligence*
(Rich dan Knight, 1991)

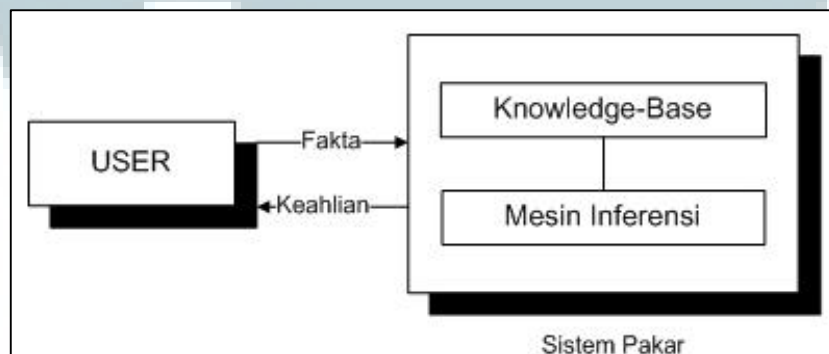
Dalam menghasilkan suatu mesin ataupun sistem “cerdas”, diperlukan teknik dasar dalam pemecahan masalah. Terdapat empat (4) teknik pemecahan masalah yang dapat dikembangkan secara terpisah ataupun dipadukan secara bersama-sama. Teknik tersebut adalah *searching*, *reasoning*, *planning*, dan *learning*. Selanjutnya, dilakukan proses penarikan simpulan atau *inferencing* berdasarkan fakta dan *rules* yang terbentuk dari metode yang digunakan.

Sistem yang dibangun pada skripsi ini tergolong pada *domain expert tasks*, dengan cabang *medical diagnosis*. Dalam membangun sistem, digunakan teknik *learning* sebagai dasar untuk memecahkan permasalahan yang ada. *Learning* merupakan teknik yang dapat secara otomatis menemukan *rules* yang diharapkan dapat berlaku umum untuk data-data lain yang belum pernah diketahui. Teknik ini digunakan ketika parameter yang digunakan begitu banyak. Kelebihan teknik *learning* adalah memungkinkan sistem sanggup untuk mempelajari setiap data baru

yang ada, sehingga *performance measure* yang dimiliki sistem juga meningkat karena pengalaman (*experience*) yang dimiliki sistem.

2.2 Sistem Pakar

Sistem pakar merupakan cabang dari AI yang memiliki dasar pengetahuan (*knowledge base*). Patterson (1990) menjelaskan, bahwa sistem pakar dapat bekerja sesuai dengan pengetahuan pakar yang dimasukkan ke dalamnya. Pengetahuan tersebut diproses menggunakan algoritma yang ada untuk menghasilkan *rules*. Selanjutnya, *rules* tersebut dijadikan dasar dalam menentukan hasil akhir dari setiap data yang diuji. Dengan demikian, sistem dapat memiliki keahlian dalam menyelesaikan permasalahan dengan mempelajari keahlian yang dimiliki oleh pakar.



Gambar 2.2 Konsep dasar sistem pakar
(Muhammad Arhami, 2006)

Konsep dasar sistem pakar dapat digambarkan seperti pada Gambar 2.2. Dapat dijelaskan sebagai berikut:

1. User

Menurut Turban (1995), terdapat tiga (3) komponen *user* yang terlibat di dalam *environment* sistem. Komponen *user* tersebut adalah:

- a. Pakar, yaitu seorang ahli yang dipercayai pendapatnya karena memiliki pengetahuan khusus dan spesifik, serta memiliki pengalaman dalam

mengaplikasikan keahliannya dalam memecahkan masalah dalam bidang yang berkaitan dengan kebutuhan sistem.

b. Perakayasa sistem (*Knowledge Engineer*), yaitu seorang *engineer* yang akan menganalisis apa yang dibutuhkan sistem, lalu mengumpulkan data berupa fakta dari pakar yang dibutuhkan dalam pembangunan sistem, kemudian mengimplementasikan area permasalahan, menginterpretasikan dan mengintegrasikan jawaban atau diagnosa yang telah dilakukan oleh pakar ke dalam sebuah sistem.

c. Pengguna (*User*), yaitu orang awam yang menggunakan sistem.

Dalam membangun sistem ini, penulis bertindak sebagai perakayasa sistem. Pakar yang terlibat pada sistem ini adalah dokter spesialis mata. Pengguna sistem ini adalah masyarakat awam.

2. Knowledge Base

Knowledge base berisi pengetahuan yang dapat digunakan dalam mendapatkan hasil dari sistem. Terdapat dua (2) bentuk pendekatan basis pengetahuan yang digunakan dalam membangun sistem, yaitu:

a. Penalaran berbasis aturan (*Rule-based reasoning*)

Pengetahuan pakar yang telah dikumpulkan akan diubah menjadi berbagai aturan dengan menggunakan *statement IF-THEN-ELSE*. *Statement* ini digunakan untuk mengolah sejumlah pengetahuan pakar dalam suatu permasalahan tertentu, kemudian sistem akan menyelesaikan masalah tersebut secara berurutan sesuai dengan aturan yang didefinisikan.

b. Penalaran berbasis kasus (*Case-based reasoning*)

Basis pengetahuan akan berisi solusi dari yang telah dicapai sebelumnya. Solusi tersebut akan dijadikan acuan untuk mendapatkan solusi baru pada keadaan yang akan diuji.

Dalam membangun sistem ini dilakukan pendekatan berbasis kasus, dimana sistem akan menerima pengetahuan beserta solusi yang telah disimpulkan oleh pakar sebelumnya sebagai fakta. Kemudian, fakta-fakta yang terkumpul tersebut dijadikan acuan dalam membentuk *rules* yang dapat menghasilkan solusi pada permasalahan *user* selanjutnya.

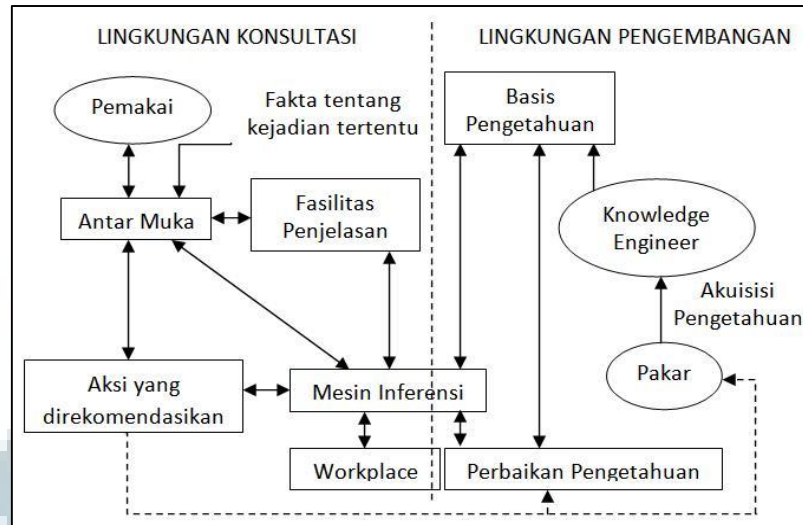
3. Mesin Inferensi

Mesin inferensi mengandung mekanisme pola pikir dan penalaran yang digunakan oleh pakar dalam menyelesaikan suatu permasalahan.

Sistem pakar yang dibangun pada skripsi ini, dapat menarik simpulan (*inferencing*) berdasarkan *rules*. Dalam membentuk *rules*, dilakukan pengumpulan fakta-fakta yang berasal dari pakar terlebih dahulu. Fakta-fakta yang terkumpul diolah menggunakan metode *learning*, sehingga dapat menghasilkan suatu keputusan yang diharapkan.

Environment yang berada di sekitar sistem pakar dapat dibagi menjadi dua (2) bagian, yaitu lingkungan pengembangan (*development environment*) dan lingkungan konsultasi (*consultation environment*) yang dapat dilihat pada Gambar

2.3. Lingkungan pengembangan sistem digunakan untuk memasukkan pengetahuan pakar, sementara lingkungan konsultasi digunakan oleh *user* untuk memperoleh pengetahuan pakar melalui sistem tersebut.



Gambar 2.3 Struktur dari sistem pakar (Muhammad Arhami, 2006)

Penggunaan sistem pakar memiliki keuntungan bagi orang awam yang menggunakannya. Keuntungan tersebut dapat dilihat dari beberapa faktor berikut:

1. Waktu

Sistem pakar dapat digunakan kapanpun pengguna ingin menggunakan sistem tersebut, sementara seorang pakar tidak selalu ada untuk dapat melakukan diagnosa terhadap penderita. Selain itu seorang pakar hanya dapat melakukan diagnosa secara satu per satu, sementara sistem pakar memungkinkan penggunaan secara serentak dimana saja pengguna berada.

2. Letak geografis

Sistem pakar dapat digunakan dimanapun pengguna berada. Hal ini karena sistem pakar berbentuk *software application* atau berupa sebuah *website* yang dapat diakses melalui jaringan internet. Hal ini berbeda dengan seorang pakar yang hanya dapat ditemui di satu tempat pada waktu tertentu.

3. Keamanan

Sistem pakar memiliki *security* untuk dapat menentukan siapa saja yang memiliki hak akses untuk menggunakannya. Simpulan yang dihasilkan oleh sistem terhadap suatu permasalahan juga akan lebih obyektif dibandingkan dengan seorang pakar yang memiliki pemikiran pribadi dalam mengambil keputusan.

4. Performance dan hasil diagnosa

Sistem pakar selalu menyimpan *knowledge* di dalam ruang penyimpanan pada sistem yang didukung dengan proses *maintenance* yang tepat guna. Sistem pakar dapat menghasilkan hasil diagnosa yang konkrit dan tidak berubah-ubah. Jika dibandingkan dengan seorang pakar dapat saja memberikan jawaban yang tidak konsisten di kemudian hari. Selain itu, seorang pakar dapat mengalami kemunduran dalam hal *knowledge* karena adanya faktor usia yang menandakan bahwa seorang pakar juga adalah seorang manusia yang dapat melakukan kesalahan.

5. Kecepatan diagnosa

Kecepatan diagnosa yang dimiliki oleh sistem pakar dalam mendiagnosa relatif lebih cepat dibandingkan dengan seorang pakar.

6. Biaya yang dikeluarkan pengguna

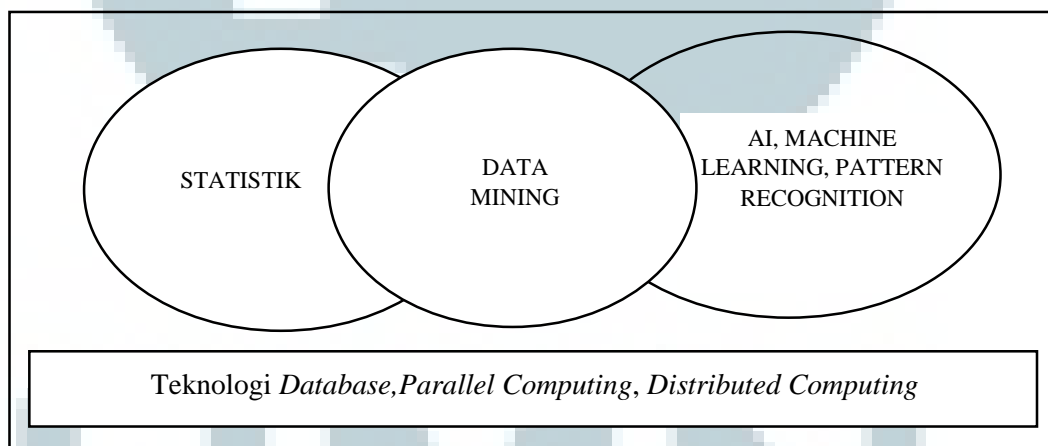
Biaya yang dibutuhkan dalam membayar seorang pakar untuk memberikan pendapat akan jauh lebih mahal, dibanding dengan biaya penggunaan sistem pakar.

2.3 Data Mining

Data mining adalah proses yang mempekerjakan satu atau lebih teknik pembelajaran komputer (*machine learning*) untuk menganalisis dan mengekstraksi pengetahuan (*knowledge*) secara otomatis (Fajar A. Hermawati, 2013). Penggunaan *data mining* memiliki tujuan untuk mengetahui pola *universal* dari data-data yang

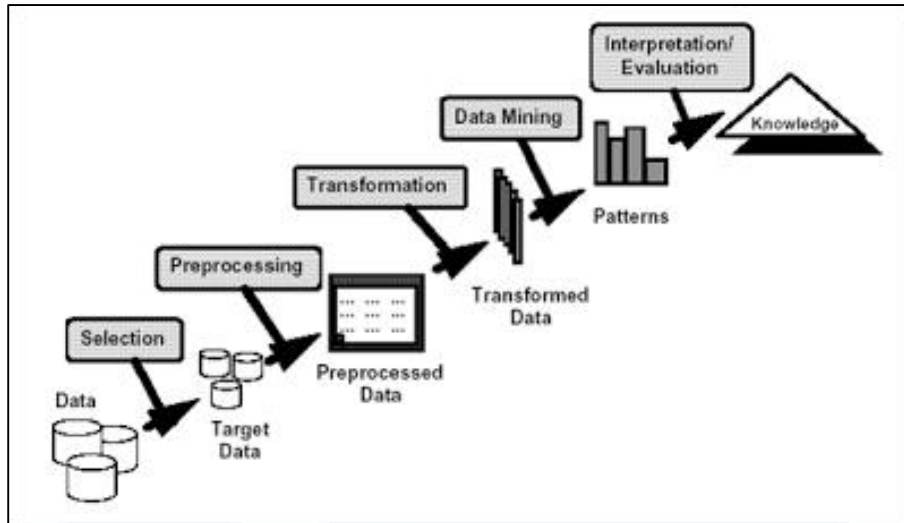
ada. Pola yang dikenali akan dijadikan bahan pokok dalam menganalisis data yang berguna dan berwawasan dalam mendukung suatu pengambil keputusan dalam suatu aktivitas bisnis cerdas (*Business Intelligence*).

Data mining memiliki kesamaan dengan disiplin ilmu lain seperti kecerdasan buatan (*Artificial Intelligence*), *data warehousing*, *statistic*, *pattern recognition*, dan *machine learning*. Kesamaan tersebut terletak pada algoritma pencarian, teknik pemodelan, dan teori pembelajarannya. Namun, tidak seratus persen sama satu sama lain. Misalnya, ilmu statistik dengan *data mining*. Dalam disiplin ilmu statistik pemodelan dibuat oleh para ahli statistik, sedangkan pada *data mining* ilmu statistik hanya dikembangkan untuk dapat menangani data yang berjumlah besar secara otomatis. Posisi *data mining* dengan beberapa bidang ilmu lainnya dapat digambarkan pada gambar 2.4.



Gambar 2.4 Posisi *data mining* di antara beberapa bidang ilmu (Eko Prasetyo, 2012)

Dalam menghasilkan suatu *knowledge* dari pola-pola yang ada, diperlukan penerapan metode *scientific* yang disebut dengan *Knowledge Discovery in Database* (KDD). *Data mining* merupakan proses penambangan pengetahuan dari sekumpulan data, proses ini adalah salah satu yang termasuk dalam tahapan proses KDD.



Gambar 2.5 Tahapan proses KDD
(Fajar A. Hermawati, 2013)

Dorian Pyle pada bukunya *Data Preparation for Data Mining*, mengestimasi bahwa tahapan persiapan data pada KDD sebelum dilakukan *mining* membutuhkan 60% dari keseluruhan waktu pemrosesan *data mining* (Larose, 2005). Penjelasan dari setiap tahap pada proses KDD adalah sebagai berikut (Fajar A. Hermawati, 2013):

1. Selection

Pada tahap pertama dilakukan seleksi data dari *multiple data source* dengan membersihkan data yang memiliki *noise* ataupun *missing value*. Tahapan pertama ini dilakukan untuk mengetahui dan menggali pengetahuan awal dari data yang terkumpul.

2. Preprocessing

Pada tahap kedua dilakukan penggabungan terhadap seluruh *data source* yang telah terkumpul sebelum tahapan dari proses selanjutnya dilanjutkan.

3. Transformation

Pada tahapan ketiga dilakukan transformasi data ke dalam bentuk yang lebih sesuai untuk dilakukan *mining*.

4. Data Mining

Tahapan ini merupakan proses terpenting, dimana suatu metode *data mining* diterapkan pada sistem, sehingga sistem dapat menghasilkan pola dari data yang terkumpul. Penerapan *data mining* dilakukan dengan mengimplementasikan algoritma dari metode *data mining* yang telah ditentukan.

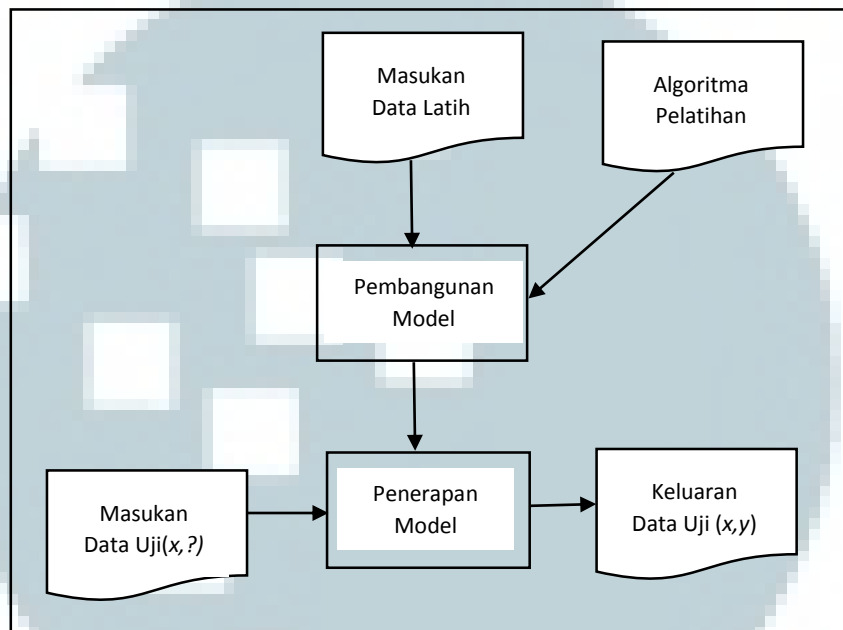
5. Interpretation / Evaluation

Tahapan selanjutnya adalah melakukan interpretasi dan evaluasi dari pola yang didapatkan, sehingga dapat diidentifikasi apakah pola tersebut sudah dapat mewakili *knowledge* yang ingin dicapai. Hasil dari evaluasi tersebut divisualisasikan untuk dapat mempresentasikan *knowledge* yang sudah didapat.

2.4 Klasifikasi

Di dalam bukunya, Larose (2005) menjelaskan beberapa teknik dan sifat *data mining*, yaitu *description, estimation, prediction, classification, clustering* dan *association*. Dari enam teknik *data mining* yang ada, teknik yang digunakan pada penelitian ini adalah klasifikasi. Klasifikasi adalah kegiatan melakukan pengelompokan data menjadi berbagai kelas data dengan tujuan memprediksikan kelas untuk data yang tidak diketahui kelasnya. Klasifikasi juga didefinisikan sebagai pekerjaan yang melakukan pelatihan terhadap fungsi target f yang memetakan setiap set atribut (fitur) x ke satu dari sejumlah *output* y yang tersedia. Dalam melakukan pengklasifikasian terdapat dua (2) tahap utama yang harus dilakukan. Tahap pertama adalah tahap induksi yaitu pembangunan model sebagai

prototype untuk disimpan sebagai memori. Tahap kedua adalah deduksi yaitu penggunaan model yang dihasilkan untuk melakukan klasifikasi, pengenalan, prediksi pada suatu objek data lain agar diketahui termasuk kelas manakah objek data tersebut tergolong di dalam memori (Eko Prasetyo, 2012).



Gambar 2.6 Proses klasifikasi (Eko Prasetyo, 2012)

Dalam menggunakan teknik klasifikasi diperlukan suatu algoritma pelatihan (*learning algorithm*) untuk mendapatkan suatu model yang paling memenuhi hubungan antara himpunan nilai dari setiap *variable* dengan hasil (*result*). Terdapat banyak sekali metode yang dapat melakukan *generate* hasil dari *set-data* yang dimasukkan. Metode *learning* yang diteliti pada skripsi ini adalah *decision tree*.

2.5 Decision Tree

Decision Tree adalah sebuah struktur data yang terdiri dari kumpulan simpul (*node*) yang saling terhubung melalui rusuk (*edge*). Simpul pada sebuah pohon dibedakan menjadi simpul akar (*root*), simpul percabangan (*branch*), dan

simpul daun (*leaf*). *Decision tree* dapat merepresentasikan secara sederhana teknik klasifikasi untuk sejumlah kelas berhingga, dimana simpul cabang maupun simpul akar ditandai dengan nama *variable* (atribut), sementara setiap rusuknya diberi label nilai dari *variable* tersebut. Atribut menyatakan suatu parameter yang dibuat sebagai kriteria dalam pembentukan *tree*. Setiap atribut memiliki nilai yang disebut dengan *record* (Fajar A. Hermawati, 2013).

Decision Tree terdiri dari sekumpulan aturan untuk membagi populasi *heterogen* menjadi *homogen* dengan memperhatikan *variable* tujuannya. Kebutuhan yang harus dipenuhi sebelum sebuah algoritma *decision tree* diterapkan adalah (Larose, 2005):

1. Algoritma *decision tree* harus merepresentasikan suatu pembelajaran (*learning*).
2. *Set-data training* harus merepresentasikan banyak kemungkinan yang dapat terjadi agar memungkinkan mendapat hasil dalam segala hal yang diprediksikan.
3. Atribut katagorikal, dari *set-data* yang digunakan dapat memiliki tipe data nominal, sementara atribut numerikal dari *set-data* yang digunakan harus merupakan atribut diskrit.

Dalam melakukan proses penarikan solusi menggunakan *decision tree* tahapan pertama yang harus dilakukan adalah mengelompokkan *set-data* pada sebuah tabel, lalu mengubah dari bentuk tabel menjadi bentuk pohon dengan menggunakan algoritma yang ditentukan. Setelah itu, ubah model pohon menjadi *rules* (Basuki dan Syarif, 2003). Tujuan melakukan konstruksi terhadap model klasifikasi tidak hanya untuk menghasilkan *rules* yang dapat menghasilkan suatu

prediksi yang akurat, meskipun hal tersebut memang menjadi konsentrasi utama (Quinlan, 1992).

Terdapat banyak algoritma *learning* yang dapat digunakan dalam pembangunan *decision tree*, diantaranya adalah ID3, C4.5, dan CART. Pada penelitian ini digunakan algoritma C4.5.

2.6 Algoritma C4.5

Algoritma C4.5 merupakan pengembangan dari algoritma ID3 yang dilakukan Quinlan (1992). Algoritma C4.5 dapat menyelesaikan masalah secara sistematis dengan cara membentuk suatu *decision tree* dengan langkah-langkah sebagai berikut:

1. Pilih atribut sebagai akar,
2. Buat cabang untuk masing-masing *record* dari atribut,
3. Membagi kasus ke dalam cabang,
4. Ulangi proses untuk masing-masing cabang sampai semua kasus pada cabang menghasilkan suatu keputusan yang sesuai.

Untuk memilih sebuah atribut menjadi akar, dilakukan perhitungan nilai dari atribut-atribut yang ada. Nilai *gain* yang paling tinggi akan dijadikan *root* pada pohon keputusan.

Untuk menghitung dapat digunakan rumus

$$Gain(S, A) = Entropy(S) - \sum_{i=1}^n i \frac{|S_i|}{|S|} \times Entropy(S_i) \dots Rumus 2.1$$

dengan:

- S : himpunan kasus
A : atribut
n : jumlah partisi atribut A

|S_i| : jumlah kasus pada partisi ke-i

|S| : jumlah kasus dalam S

Sedangkan untuk perhitungan nilai *entropy* dapat dilakukan dengan rumus

$$Entropy(S) = - \sum_{i=1}^n p_i \times \log_2 p_i \quad \dots \text{ Rumus 2.2}$$

dengan:

S : himpunan kasus

n : jumlah partisi S

p_i : proporsi dari S_i terhadap S

Setelah perhitungan selesai, maka dilakukan pembentukan pohon keputusan dilanjutkan dengan pembentukan *rules* menggunakan kaidah IF-THEN-ELSE.

2.7 Katarak

Katarak berasal dari Bahasa Yunani, *Katarrhakies* yang berarti air terjun. Katarak merupakan suatu kelainan pada mata. Kelainan ini bukan suatu tumor atau tumbuhnya jaringan di dalam mata, melainkan suatu keadaan dimana lensa mata yang terlihat jernih akan berubah menjadi keruh (Ilyas Sidarta dkk., 2002). Kekeruhan pada serabut atau bahan lensa di dalam kapsul lensa dapat terjadi pada saat perkembangan serat lensa masih berlangsung atau sesudah serat lensa berhenti dalam perkembangannya dan mulai memasuki tahap degenerasi (Ilyas Sidarta, 2001).

Proses kekeruhan lensa mata dapat terjadi akibat hidrasi lensa, denaturasi protein lensa atau akibat gabungan dari keduanya. Kekeruhan ini terjadi akibat gangguan metabolisme normal lensa yang dapat timbul di berbagai tingkat usia. Pada umumnya, kekeruhan lensa pada kedua mata dapat terjadi secara progresif atau bahkan tidak mengalami perubahan apapun dalam waktu yang lama (Ilyas

Sidarta dkk., 2002). Apabila tidak ada penanganan apapun yang dilakukan, maka kekeruhan lensa mata ini akan semakin tebal dan memiliki resiko fatal kebutaan pada mata.

2.7.1 Penyebab Katarak

Terdapat berbagai faktor yang dapat menyebabkan seorang manusia mengalami katarak. Penyebab tersebut dijelaskan pada bagian ini.

1. Usia

Katarak dapat terjadi karena proses regeneratif atau bertambahnya usia manusia. Semakin bertambahnya usia, lensa juga mengalami perkembangan dan bertambah tua, sehingga pada umumnya katarak dialami oleh para lansia (lanjut usia). Meski usia lanjut menjadi salah satu faktor penyebab katarak secara umum, tetapi katarak juga ditemukan pada manusia yang berusia di luar batas minimal usia para lansia (Ardiansah, 2011).

2. Cedera Mata

Riwayat terjadinya cedera pada mata, dapat membuat timbulnya rasa traumatik pada mata manusia. Adanya trauma menyebabkan kerusakan secara langsung pada protein lensa, sehingga menimbulkan katarak.

3. Penyakit sistemik

Adanya riwayat penyakit sistemik yang dimiliki oleh penderita dapat menyebabkan perubahan metabolisme pada lensa yang kemudian menimbulkan terjadinya katarak. Penyakit yang dimaksud antara lain *diabetes mellitus*, *hipoparatiroid*, *down syndrome*, dan *dermatitis atopic*.

4. Radiasi sinar ultraviolet

Terlalu sering berada dalam keadaan terpapar sinar matahari, meningkatkan resiko terjadinya komplikasi kesehatan akibat paparan radiasi sinar ultraviolet. Radiasi ultraviolet dapat meningkatkan jumlah radikal bebas pada lensa mata, sehingga beresiko dalam menimbulkan katarak.

5. Campak (infeksi Virus rubella)

Katarak akibat campak dapat dialami oleh bayi yang terinfeksi Virus rubella pada saat seorang ibu memasuki kehamilan tri semester pertama.

6. Riwayat keturunan

Adanya anggota keluarga inti yang telah mengalami katarak, meningkatkan resiko seseorang mengalami katarak, serta memiliki kemungkinan pematangan katarak yang lebih cepat terjadi pada orang tersebut.

7. Riwayat penggunaan obat *steroid*

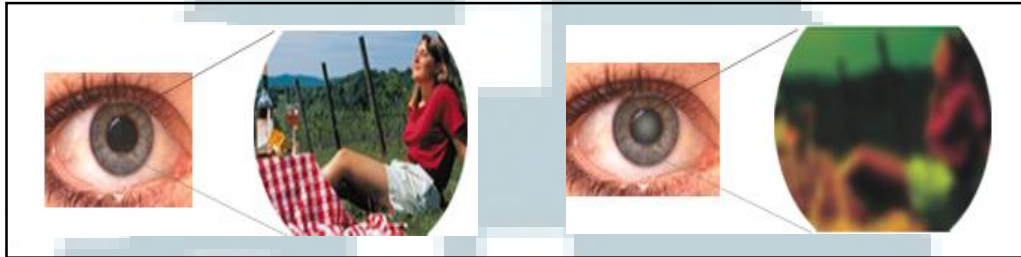
Adanya infeksi atau *uveitis anterior* yaitu peradangan yang terjadi pada mata, membuat penderita harus menggunakan obat *steroid* untuk mengatasinya. Resiko penggunaan obat *steroid* dengan dosis tinggi dalam jangka panjang dapat menimbulkan terjadinya katarak. Hal ini karena obat *steroid* dapat menyebabkan terjadinya denaturasi dari protein.

2.7.2 Gejala Katarak

Gejala yang dapat dialami penderita katarak disebabkan karena adanya proses kekeruhan yang terjadi pada lensa mata. Gejala tersebut adalah

1. Pandangan mata kabur
2. Perubahan dalam persepsi warna
3. Penglihatan ganda pada satu mata (*Halo*)

4. Penglihatan di malam hari lebih berkurang
5. Penglihatan menguning
6. Seperti ada titik gelap di depan mata
7. Susah untuk melihat di tempat yang terang karena silau
8. Pandangan yang kurang tajam.



Gambar 2.7 Perbedaan pandangan pada mata normal (kiri) dengan mata katarak (kanan)
(www.google.com)

2.7.3 Jenis Katarak

Ilyas Sidarta (2001) juga mengklasifikasikan jenis katarak berdasarkan penyebab-penyebab katarak yang telah disebutkan di atas. Pada bagian ini dijelaskan mengenai jenis dari katarak

1. Katarak Primer

Katarak primer adalah katarak yang diklasifikasikan berdasarkan usia.

a. Katarak Kongenital

Katarak yang terlihat pada usia di bawah 1 tahun.

b. Katarak Juvenil

Katarak yang terjadi di atas usia 1 tahun dan di bawah 40 tahun.

c. Katarak Presenilis

Katarak yang terjadi pada penderita berusia 30 – 40 tahun.

d. Katarak Senilis

Katarak yang terjadi pada penderita berusia lebih dari 40 tahun.

Pada katarak primer, kekeruhan lensa yang dialami penderita tidak terjadi dalam waktu singkat, sehingga gejalanya tidak muncul secara mendadak. Maka, penderita katarak primer awalnya sangat sering melakukan pergantian kacamata dan menganggap gejala tersebut hanya gangguan mata biasa.

2. Katarak Sekunder

Katarak sekunder dapat terjadi pada penderita katarak yang telah menjalani bedah lensa dan terjadi reaksi radang yang berakhir dengan terdapatnya jaringan *fibrosis* sisa lensa yang tertinggal. Tindakan bedah yang dapat menimbulkan katarak sekunder adalah sisa *disisio lentis*, ekstraksi linear dan ekstraksi lensa ekstrakapsular (Ilyas Sidarta, 2001).

3. Katarak Komplikata

Katarak komplikata adalah katarak yang dapat terjadi akibat adanya komplikasi dari penyakit lain yang dialami oleh penderita sebelumnya. Penyakit tersebut antara lain gangguan pada bagian mata yang lain, seperti *ablasio retina* yang sudah lama, *retinitis pigmentosa glaucoma*, *myopia maligna*, dan *uveitis*. Selain itu dapat juga dikarenakan penderita yang mengalami penyakit sistemik, seperti *diabetes mellitus*, *hipoparatiroid*, *down syndrome*, dan *dermatitis atopik*. Katarak komplikata juga dapat disebabkan karena adanya trauma terhadap pukulan, masuknya benda asing ke dalam mata, terpapar panas api, sinar matahari, sinar-x, radioaktif, dan toksis kimia secara berlebihan (Ilyas Sidarta, 2001).

4. Katarak Kongenital

Katarak kongenital adalah katarak yang dialami oleh bayi pada waktu lahir. Kekeruhan yang terjadi pada lensa bayi ini sudah terjadi sejak lensa baru saja mengalami pembentukan. Katarak ini sering ditemukan pada bayi yang dilahirkan oleh ibu yang menderita *rubella*, *diabetes mellitus*, *toksoplasmosis*, *hipoparatiroidisme*, ataupun *galaktosemia* (Ilyas Sidarta, 2001). Dengan melihat adanya batasan masalah, bahwa sistem ini hanya dapat digunakan oleh pengguna dewasa, maka katarak tipe ini tidak menjadi keluaran sistem.

2.7.4 Pengobatan Katarak

Bagi penderita katarak stadium awal (*insipient*), penderita dapat menggunakan kacamata sebagai bantuan untuk menjalankan kegiatan sehari-hari dan tidak membutuhkan tindakan pembedahan. Namun, bagi penderita yang kataraknya sudah terasa sangat mengganggu, diperlukan tindakan operasi (Ilyas Sidarta, 1989). Pembedahan katarak terdiri atas:

1. Pengangkatan lensa

Dua (2) jenis pembedahan dalam pengangkatan lensa adalah:

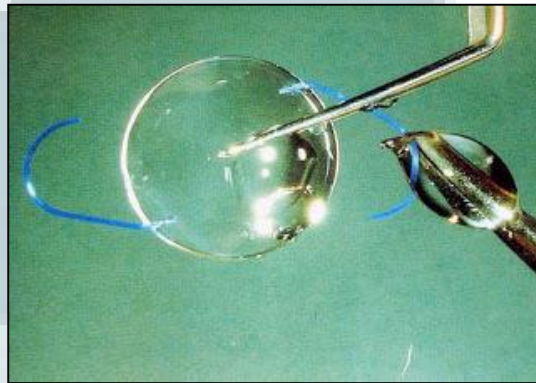
a. Pembedahan *Extra Capsular Cataract Extraction (ECCE)*

Lensa diangkat dengan meninggalkan *capsule*. Tindakan ini memiliki tujuan untuk memperlunak lensa, sehingga mempermudah pengambilan lensa melalui sayatan kecil. Pembedahan yang disebut sebagai *phacoemulsification* yaitu pengambilan lensa menggunakan gelombang suara berfrekuensi tinggi (Ilyas Sidarta, 1989).

b. Pembedahan *Intra Capsular Cataract Extraction (ICCE)*

Lensa diangkat beserta *capsule*. Pada saat ini, pembedahan jenis ini sudah jarang dilakukan (Ilyas Sidarta, 1989).

2. Penggantian lensa



Gambar 2.8 Lensa tanam *intraocular*
(www.google.com)

Setelah melakukan pembedahan, lensa penderita katarak akan diganti dengan lensa *intraocular*. Lensa ini adalah lensa buatan berupa lempengan plastik yang ditanam ke dalam kapsul lensa di dalam mata. Setelah mengetahui penyebab dan gejala yang dialami penderita, maka penderita harus melakukan tindakan medis secepatnya. Apabila tidak diobati akan membuat penglihatan penderita semakin bertambah gelap, memiliki resiko katarak menjadi pecah dan menimbulkan komplikasi lain yang cukup berat, seperti *glaucoma* dan peradangan dalam bola mata (Ilyas Sidarta, 1989).

2.8 Bootstrap

Bootstrap merupakan *framework* yang dapat digunakan untuk membangun tampilan *responsive website* secara mudah, cepat, dan gratis. Bootstrap menggunakan *960 grid system*, dengan lebar maksimal *grid* 940px dan jumlah kolom sebanyak 12 kolom pada layar *desktop* (Husein Alatas, 2013). Bootstrap

dibangun pertama kali oleh Mark Otto dan Jacob Thornton pada pertengahan tahun 2010. Keduanya adalah *programmer social media* Twitter. Pada 19 Agustus 2011, Bootstrap secara resmi dirilis dan dapat digunakan secara *open source* oleh para pengembang *website*.

Versi yang digunakan dalam pembangunan *website* ini adalah Bootstrap v2.3.2. Keuntungan menggunakan Bootstrap adalah dapat memudahkan dan memberikan efisiensi lebih kepada pengembang dalam membangun *website* yang tampilannya dapat kompatibel dengan ukuran layar *handphone*, *tablet*, maupun *desktop PC*. Bootstrap juga menyediakan fitur *JQUERY plugin* yang dapat mempercantik tampilan *website*. Di antaranya adalah *dropdown*, *tooltip*, *modal*, *button*, *alert* dan masih banyak lagi. Untuk menggunakan *framework* ini, seluruh dokumentasi dapat di *download* pada situs resmi Bootstrap <http://getbootstrap.com>.

2.9 K-fold Cross Validation

Validasi berkaitan dengan penerapan *set-data* yang berbeda untuk memastikan bahwa *software* yang telah dibangun sesuai dengan *requirement* awal dari pengembangan sistem. Salah satu teknik validasi yang dapat digunakan dalam klasifikasi adalah *k-fold cross validation*. Teknik ini merupakan teknik umum yang digunakan dalam mengestimasi performa dari teknik pengklasifikasian pada jumlah *set-data* yang kecil (Kohavi, 1995).

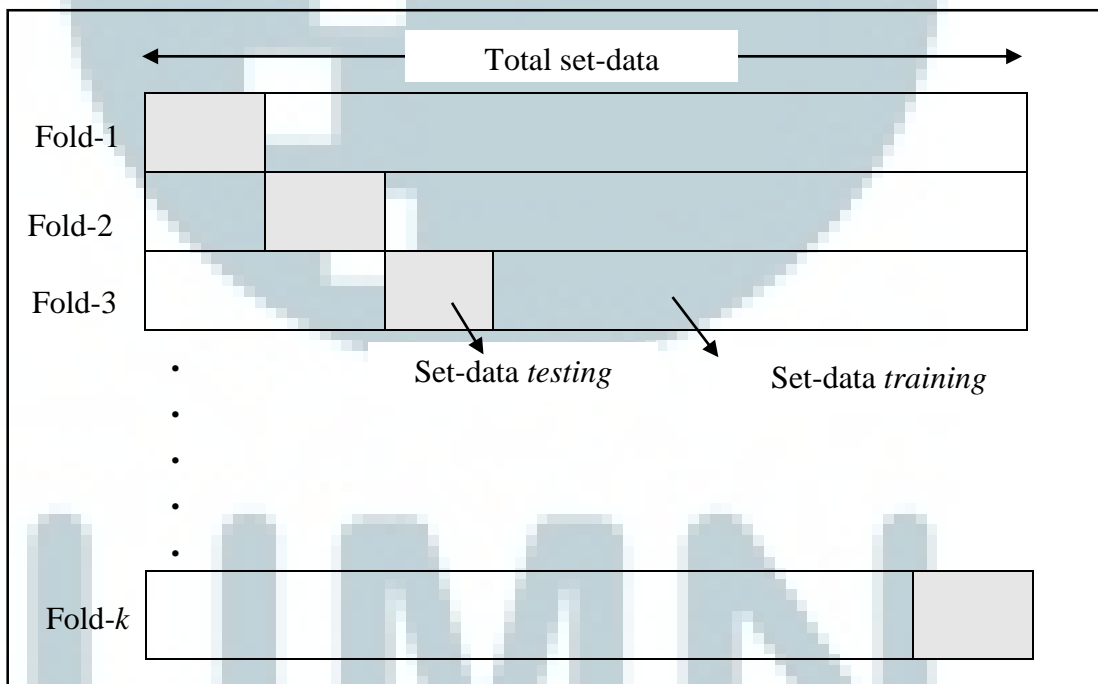
Cara untuk melakukan teknik *k-fold cross validation* adalah dengan menentukan nilai *k* terlebih dahulu. Nilai *k* digunakan sebagai ukuran pembagi *set-data* yang diuji. Penggunaan nilai *k* terbaik untuk menguji validitas data *training* adalah sebesar 10 (Breiman dan Spector, 1992). Setelah itu, lakukan langkah-langkah berikut:

1. Bagi *set-data* ke dalam $k=10$ bagian,
2. Jadikan *fold* ke-1 sebagai data *testing* dan *fold* ke-2 hingga 10 sebagai data *training*,
3. Lakukan *testing* dengan data *testing* tersebut,
4. Hitung nilai akurasi yang didapatkan,

$$CV(\lambda) = \frac{1}{K} \sum_{k=1}^K E_k(\lambda) \quad \dots \text{ Rumus 2.3}$$

5. Ulangi langkah ke-2 sampai ke-4 hingga *fold* ke-10,
6. Hitung rata-rata akurasi dari 10 *fold* tersebut.

Gambaran lebih jelasnya dapat digambarkan pada Gambar 2.9 (Hastie, Tibshirani dan Friedman, 2008).



Gambar 2.9 *k-fold cross validation*