

# ***Kecerdasan Buatan*** **Kajian Konsep dan Penerapan**



**Jarot Dian Susatyo, M.Kom**



# ***Kecerdasan Buatan***

## **Kajian Konsep dan Penerapan**



**Jarot Dian Susatyo, M.Kom**

# ***KECERDASAN BUATAN***

**Kajian Konsep dan Penerapan**

**Jarot Dian Susatyono, M.Kom**



**YAYASAN PRIMA AGUS TEKNIK**

## **KECERDASAN BUATAN, Kajian Konsep dan Penerapan**

### **Penulis:**

Jarot Dian Susatyo, M.Kom

**ISBN:** 978-623-6141-31-1 (PDF)

### **Editor:**

Indra Ava Dianta, S.Kom., M.T

### **Penyunting:**

Yuli Fitrianto, ST., M.Kom

### **Desain Sampul dan Tata Letak:**

Andik Prakasa Hadi, M.Kom

### **Penerbit:**

Yayasan Prima Agus Teknik

Redaksi:

Jln Majapahit No 605 Semarang

Tlpn. (024) 6723456

Fax . 024-6710144

Email: penerbit\_ypat@stekom.ac.id

### **Distributor Tunggal:**

UNIVERSITAS STEKOM

Jalan Majapahit No. 605 Semarang

Tlpn. (024) 6723456

Fax. 024-6710144

Email: info@stekom.ac.id

Hak Cipta dilindungi Undang-Undang

Dilarang memperbanyak karya tulis ini dalam bentuk dan dengan cara apapun tanpa ijin tertulis dari penerbit.

# HALAMAN PENGESAHAN

Judul Buku Ajar : Kecerdasan Buatan Kajian Konsep dan Penerapan  
Mata Kuliah : Kecerdasan Buatan  
Nama Penulis : Jarot Dian Susatyono, M.Kom  
NIDN : 0616018301  
Progdi : Sistem Komputer  
Fakultas : Fakultas Komputer dan Bisnis (FKB)  
Unit : Universitas Sains dan Teknologi Komputer

Semarang, 26 April 2021

Menyetujui

Kaproghi S1-Sistem Komputer

Ketua Penerbit YPAT

**Iman Saufik Suasana, M.Kom**  
NIDN: 0609097602

**Danang, S.Kom., M.T**  
NIDN: 0615098702

# KATA PENGANTAR

Rasa syukur tak lupa penulis panjatkan kehadirat Allah SWT atas berkat, rahmat dan karunia-Nya sehingga buku ajar untuk mata kuliah Kecerdasan Buatan ini telah berhasil diselesaikan. Buku ajar ini diperuntukkan sebagai pegangan bagi mahasiswa program studi Sistem Komputer yang dirancang secara praktis agar mahasiswa dapat memahami dan mempraktekkan mulai dari bagaimana menginterpretasikan konsep kecerdasan yang dimiliki oleh manusia ke aplikasi komputer.

Buku ajar ini dibuat dengan maksud memberikan materi pengajaran kecerdasan buatan yang lengkap dan praktis untuk mahasiswa sesuai dengan standar kurikulum yang telah ditentukan dan disepakati bersama. Materi dan tugas pengajaran dikembangkan dengan model tutorial yang telah terintegrasi sehingga dapat berkembang kompetensi mahasiswa.

Harapan ke depannya, buku ajar ini merupakan bahan dan sumber belajar dan kompetensi mahasiswa yang sesuai sebagai bekal di dunia kerja nantinya, maka dari itu agar isi dari buku ajar ini selalu relevan dengan kondisi terkini dengan dinamika perubahan teknologi yang terus terjadi, selalu dibutuhkan masukan untuk bahan perbaikan atau revisi.

Demikian, sehingga buku ajar ini dapat memberi manfaat untuk kita bersama dan khususnya mahasiswa/mahasiswi UNIVERSITAS STEKOM Semarang atau praktisi yang sedang mengembangkan bahan ajar untuk kegiatan perkuliahan.

Semarang, 26 April 2021

Penulis

# DAFTAR ISI

	Halaman
Bab 1 : Mengenal Kecerdasan Buatan .....	1
Bab 2 : Algoritma Pencarian ( <i>Searching Algorithm</i> ).....	19
Bab 3 : Representasi Pengetahuan.....	34
Bab 4 : Sistem Pakar.....	44
Bab 5 : Probabilitas Markov .....	86
Bab 6 : Probabilitas Bayesian .....	97
Bab 7 : First Order Logic .....	104
Bab 8 : Logika Samar (Fuzzy Logic).....	110
Bab 9 : Pengolahan Bahasa Alami ( <i>Natural Language Processing</i> )	118
Bab 10 : Game Playing .....	128
Bab 11 : Algoritma Genetika.....	132
Bab 12 : Computer Vision .....	136
Bab 13 : Robotika.....	143
Bab 14 : Jaringan Saraf Tiruan ( <i>Artificial Neural Network</i> ).....	151

# BAB I

## Mengenal Kecerdasan Buatan

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang evaluasi kecerdasan buatan
  - ❖ Menjelaskan konsep dan definisi kecerdasan buatan
  - ❖ Menjelaskan tujuan kecerdasan buatan
  - ❖ Menjelaskan perbedaan kecerdasan alami dan kecerdasan buatan
  - ❖ Menjelaskan penerapan kecerdasan buatan
  - ❖ Menjelaskan soft computing.
- 

### A. Evolusi Kecerdasan Buatan

Manusia adalah makhluk yang cerdas dalam memecahkan problem karena manusia memiliki pengetahuan dan pengalaman. Pengetahuan diperoleh dalam proses pembelajaran. Semakin banyak pengetahuan yang dimiliki seseorang, semakin kuat kemampuannya untuk memecahkan problem. Namun, pengetahuan saja tidak cukup, manusia juga memiliki alasan untuk menarik kesimpulan berdasarkan pengetahuan dan pengalaman mereka sendiri. Manusia dengan banyak pengalaman dan pengetahuan tidak bisa berdebat dengan baik, anda tidak akan pandai memecahkan problem. Keterampilan dan pengetahuan tidak bisa

menyelesaikan problem dengan baik. Demikian pula, manusia yang memiliki kemampuan berpikir yang kuat tetapi tidak memiliki pengetahuan dan pengalaman yang benar tidak dapat menyelesaikan problem dengan benar.

Pada awal abad ketujuh belas, Rene Descartes percaya bahwa tubuh hewan tidak lebih dari mesin yang rumit. Byes Pascal menciptakan komputer digital mekanis pertama pada tahun 1642. Pada usia 19 tahun, Charles Babbage dan Ada Lovela menggunakan komputer mekanis yang dapat diprogram. Bertrand Russell (Bertrand Russell) dan Alfred North Whitehead (Alfred North Whitehead) menerbitkan "Principles of Mathematics", yang merevisi logika formal. Warren McCulloch dan Walter Pitts menerbitkan "Kalkulus Logis dari Pemikiran Konstan" pada tahun 1943, yang meletakkan dasar bagi jaringan saraf. 1950-an adalah upaya tak henti-hentinya di bidang kecerdasan buatan. Program kecerdasan buatan operasional pertama ditulis pada tahun 1951 untuk menjalankan mesin Ferranti Mark I di program permainan catur Universitas Manchester: Christopher Strachey dan Dietrich Prinz Prinz).

John McCarthy menciptakan istilah "kecerdasan buatan" pada pertemuan pertama tentang problem ini pada tahun 1956. Dia juga menemukan bahasa pemrograman Lisp. Alan Turing (Alan Turing) mengusulkan "Tes Turing" untuk menguji perilaku cerdas. Joseph Weizenbaum mengembangkan ELIZA, pembongkaran yang menggunakan psikoterapi Roger. Pada 1960-an dan 1970-an, Joel Moses mendemonstrasikan kemampuan penilaian simbolis untuk memecahkan problem. Program matematika berbasis pengetahuan pertama yang berhasil di Macsyma. Marvin Minsky dan Seymour Papert menerbitkan perceptrons yang membuktikan keterbatasan jaringan saraf sederhana, dan Alain Colmerau mengembangkan bahasa komputer Prolog. Dr. Med Shortliffe telah

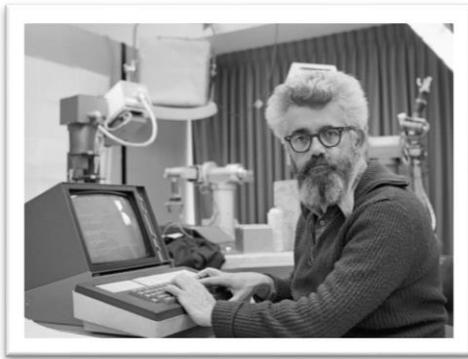
mendemonstrasikan kemampuan sistem berbasis aturan untuk menampilkan dan memperoleh pengetahuan tentang diagnosis dan pengobatan medis, dan terkadang hal itu dapat dilakukan.

Hans Moravec (Hans Moravec) dikenal sebagai sistem pakar batch pertama, yang mengembangkan kendaraan pertama yang dikendalikan komputer yang dapat mengemudi di jalan yang kompleks. Pada 1980-an, jaringan saraf banyak digunakan dalam algoritma propagasi mundur, yang pertama kali diusulkan oleh Paul John Werbo pada 1974. Pada 1990-an, berbagai bidang kecerdasan buatan membuat kemajuan besar, membuktikan penerapannya yang luas. Secara khusus, Deep Blue, komputer yang digunakan untuk permainan catur, mengalahkan Garry Kasparov dalam permainan Match 6 yang terkenal pada tahun 1997. DARPA mengatakan bahwa dalam Perang Dunia Pertama di kawasan Teluk, penghematan biaya penerapan teknologi AI ke unit perencanaan menggantikan semua investasi dalam penelitian AI. Kecerdasan buatan adalah bidang ilmiah yang relatif muda. Pada 1950-an, para ilmuwan dan peneliti ingin tahu bagaimana mesin dapat melakukan pekerjaannya seperti manusia.



## Alan Mathison Turing

Alan Mathison Turing pada tahun 1950-an, seorang matematikawan dari Inggris pertama kali mengusulkan adanya tes untuk bisa melihat bisa tidaknya sebuah mesin dikatakan cerdas. Hasil tes tersebut kemudian dikenal dengan Turing *test*, dimana si mesin tersebut menyamar seolah-olah sebagai seseorang di dalam suatu permainan yang mampu memberikan respon terhadap serangkaian pertanyaan yang diajukan. Turing beranggapan bahwa, jika mesin dapat membuat seseorang percaya bahwa dirinya mampu berkomunikasi dengan orang lain, maka dapat dikatakan bahwa mesin tersebut cerdas (seperti layaknya manusia).

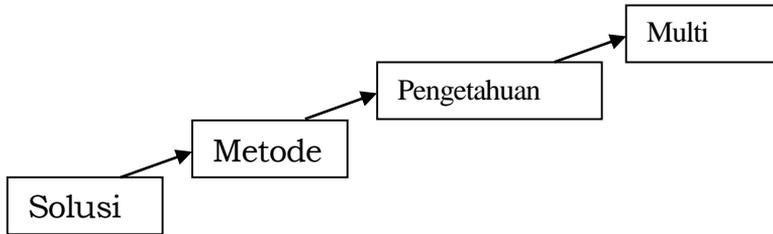


## John McCarthy

Kecerdasan buatan atau *artificial intelligence* itu sendiri dimunculkan oleh seorang professor dari Massachusetts Institute of Technology yang bernama John McCarthy pada tahun 1956 pada *Dartmouth Conference* yang dihadiri oleh para peneliti AI. Pada konferensi tersebut juga didefinisikan tujuan utama dari

kecerdasan buatan, yaitu: mengetahui dan memodelkan proses-proses berfikir manusia dan mendesain mesin agar dapat menirukan kelakuan manusia tersebut.

Perkembangan kecerdasan buatan mencakup empat tahap utama.



Beberapa program AI yang mulai dibuat pada tahun 1956-1966, antara lain :

1. Logic Theorist, diperkenalkan pada Darmouth Conference, program ini dapat membuktikan teorema-teorema matematika.
2. Sad Sam, diprogram oleh Robert K. Lindsay (1960). Program ini dapat mengetahui kalimat-kalimat sederhana yang ditulis dalam bahasa inggris dan mampu memberikan jawaban dari fakta-fakta yang didengar dalam sebuah percakapan.
3. ELIZA, diprogram oleh Joseph Weizenbaum (1967). Program ini mampu melakukan terapi terhadap pasien dengan memberikan beberapa pertanyaan.

## **B. Konsep dan Definisi Kecerdasan Buatan**

Pada tahun 1956 John McCarthy mendeskripsikan Artificial Intelligences sebagai kemampuan untuk mencapai kesuksesan dalam menyelesaikan suatu kasus tertentu. Manusia menjadi cerdas dalam menyelesaikan permasalahan yang dihadapi karena adanya pengetahuan dari belajar dan pengalaman yang dimiliki.

Kecerdasan buatan adalah studi tentang bagaimana membuat komputer melakukan tindakan yang lebih baik dibandingkan yang dilakukan manusia. Kecerdasan buatan (artificial intelligence) merupakan salah satu bagian dari ilmu komputer yang mempelajari bagaimana membuat mesin komputer dapat melakukan seperti dan sebaik yang dilakukan manusia bahkan bisa lebih baik dibandingkan apa yang dilakukan manusia. Pada awal diciptakannya, komputer hanya digunakan sebagai mesin perhitungan. Akan tetapi seiring dengan perkembangan jaman, maka peran dari komputer semakin mendominasi kehidupan umat manusia. Komputer tidak lagi digunakan sebagai mesin perhitungan, lebih dari itu, komputer diharapkan untuk dapat diberdayakan untuk mengerjakan segala sesuatu yang bisa dikerjakan oleh manusia.

Berdasarkan *Encyclopedia Britannica* mendefinisikan Kecerdasan Buatan (AI) merupakan cabang dari ilmu komputer yang dalam merepresentasi pengetahuan sebagian besar menggunakan bentuk atribut-atribut daripada bilangan, dan memproses informasi berdasarkan metode heuristic atau dengan berdasarkan sejumlah aturan. Kecerdasan Buatan berusaha untuk membangun entitas yang cerdas serta memahaminya. Alasan lain untuk belajar AI adalah bahwa entitas cerdas yang dibangun ini menarik dan berguna. Cerdas yaitu memiliki pengetahuan serta pengalaman, penalaran (bagaimana membuat keputusan & mengambil tindakan), moral yang baik.

Menurut Alexander Simon (2001) kecerdasan buatan merupakan suatu wilayah penelitian, aplikasi dan instruksi yang berkaitan dengan pemrograman komputer dalam melaksanakan segala hal yang dalam pandangan manusia tergolong cerdas.

Rich dan Knigh (1991) mendefinisikan Artificial Intelligences yaitu suatu kajian mengenai pemanfaatan komputer dalam melakukan segala hal yang lebih baik dibandingkan tenaga manusia.

Supaya komputer dapat berperilaku seperti dan sebaik manusia, maka komputer juga harus diberi suplai pengetahuan, dan mempunyai kemampuan untuk berfikir. Maka dari itu pada Artificial Intelligences, akan mencoba untuk memberikan beberapa metoda supaya dapat membekali komputer dengan kedua komponen tersebut agar komputer bisa menjadi mesin yang cerdas.

Definisi Artificial Intelligences bisa diterjemahkan dari berbagai macam sudut pandang, diantaranya yaitu:

1. Sudut pandang kecerdasan.

Kecerdasan buatan akan menjadikan mesin komputer menjadi lebih cerdas (mampu melakukan suatu hal seperti apa yang dilakukan oleh manusia).

2. Sudut pandang penelitian.

Kecerdasan buatan adalah suatu kajian bagaimana membuat agar supaya komputer dapat mengerjakan sesuatu sebaik yang dikerjakan oleh manusia. Ranah yang sering dibahas oleh para ahli serta peneliti meliputi:

- a. *Mundane task*

- Persepsi (*vision & speech*)
- Bahasa alami (*understanding, generation, & translation*)
- Pemikiran yang bersifat *commonsense*.
- Robot *control*

- b. *Formal task*

- Permainan/*games*.

- Matematika (geometri, logika, kalkulus integral, pembuktian)

c. *Expert task*

- Analisis finansial
- Analisis medikal
- Analisis ilmu pengetahuan
- Rekayasa (desain, pencarian kegagalan, perencanaan manufaktur)

3. Sudut pandang bisnis.

Artificial Intelligences adalah sekelompok piranti yang cukup kuat dan sistematis dalam menyelesaikan permasalahan-permasalahan dalam bisnis.

4. Sudut pandang pemrograman.

Kecerdasan buatan meliputi kajian tentang bahasa pemrograman simbolik, penyelesaian problem (problem solving) dan pencarian (searching). Untuk menggunakan aplikasi kecerdasan buatan ada 2 bagian penting yang sangat dibutuhkan, diantaranya yaitu :

- a. Basis pengetahuan (knowledge base), berisi kenyataan-kenyataan, hipotesis, pemikiran dan hubungan antara satu dengan lainnya.
- b. Motor inferensi (inference engine), adalah kemampuan untuk menarik keputusan berdasarkan pengalaman.

### C. Tujuan Kecerdasan Buatan

Menurut Winston dan Prendergast (1984) tujuan dari kecerdasan buatan antara lain:

1. Membuat mesin menjadi lebih pintar
2. Memahami tujuan ilmiah kecerdasan buatan
3. Membuat mesin lebih bermanfaat

Artificial Intelligences mempunyai tujuan untuk mengembangkan metode dan sistem untuk memecahkan problem, biasanya diselesaikan melalui aktivitas intelektual manusia, misal pengolahan citra, perencanaan, peramalan dan lain-lain serta meningkatkan kinerja sistem informasi yang berbasis komputer. Sistem kecerdasan buatan juga mampu meningkatkan pengertian atau pemahaman bagaimana otak manusia bekerja.

Sedangkan menurut Lenat dan Feigenbaum (1992) terdapat sembilan tujuan kecerdasan buatan yaitu :

1. Memahami pemahaman manusia, mencoba untuk memperoleh kepandaian ingatan manusia yang mendalam, keahlian problem solving, menelaah, membuat keputusan dan lain-lain.
2. Mekanisasi biaya efektif, mengambil alih manusia dalam tugas-tugas inteligensi, memiliki program yang kinerjanya sebaik manusia dalam mengerjakan pekerjaan.
3. Pemantapan intelegensi biaya praktis, memanifestasikan sistem untuk menolong manusia berpikir semakin baik, semakin cepat, semakin dalam, dan lain-lain. Contoh: sistem pendeteksiian penyakit.
4. Intelegensia manusia hebat, menciptakan program yang memiliki kesanggupan untuk melebihi intelegensia manusia.
5. Pemecahan problem umum, sistem penanganan berbagai perkara yang luas, sistem ini mempunyai keluasan pemikiran.
6. Wacana terintegrasi, korespondensi dalam manusia mengaplikasikan bahasa alamiah, contoh : dialog, cerdas yang ada dalam Turing Test.
7. Studi (induksi), sistem sebaiknya mampu untuk mendapatkan data sendiri dan mengetahui bagaimana memperolehnya, sistem dapat

menyamarkan, membuat hipotesis, penggunaan atau penelaahan secara heuristik, membuat argumentasi dengan analogi.

8. Kemandirian, mempunyai sistem intelegensia yang bekerja atas daya usaha sendiri, harus bereaksi dengan dunia nyata.
9. Informasi, simpan informasi dan mengerti cara untuk mendapatkan informasi.

#### **D. Karakteristik Kecerdasan Buatan**

Karakteristik dari kecerdasan buatan meliputi :

1. Sebuah mekanisme yang mampu berpikir layaknya manusia, dimana keinginan awal di ciptakan konsep kecerdasana buatan yang di tanam dalam sebuah system yakni bertujuan menyelesaikan problem persis seperti manusia berpikir bahkan dengan hasil yang lebih baik dari manusia itu sendiri.
2. Sistem yang bekerja seperti manusia, Negara Asia Timur terutama Jepang merupakan salah satu kubah kemajuan teknologi dunia yang muktahir perkembangannya. Dengan menciptakan kemampuan sebuah mesin yang bertindak seperti manusia yang di terapkan dalam konsep robotic yang sangat memungkinkan bahwa generasi manusia kedepan sebagian besar para mesin – mesin robotlah yang lebih berperan.
3. Sistem yang mampu berpikir secara ilmiah. Dimana konsepsi logika yang di tanamkan mampu memilah sebuah data yang disajikan untuk melakukan proses berpikir secara mesin dengan metode keputusan yang lebih rasional. Misalkan, manusia bisa berpikir dan bertindak diluar rasional akan sebuah mesin itu sendiri tanpa ada data atau informasi, sedangkan sebuah mesin tanpa

adanya sebuah data dan informasi yang di berikan tak lebih hanyalah sebuah benda mati seperti batu.

4. Sistem mampu bertindak secara rasional, dimana konsep berpikir secara rasional di tanamkan pada mesin itu sendiri akan mampu melakukan tindakan yang tentunya secara rasional dan masuk akal. Kecerdasan buatan itu sendiri merupakan gabungan antara ilmu pengetahuan berupa pengalaman, ilmu pengetahuan, bahasa mesin serta bahasa natural(alami) yang di kosenp dalam area kecerdasan itu sendiri berupa robotic,computer, natural language, machine learning, parttern recognition, expert system, speech.

### E. Perbedaan Kecerdasan Alami dan Kecerdasan Buatan

Perbandingan sistem konvensional dan kecerdasan buatan dapat dilihat dari tabel berikut :

Kecerdasan Alami	Kecerdasan Buatan
1.Cepat mengalami perubahan. Hal ini disebabkan karena sifat manusia adalah pelupa.	1.Bersifat permanen, selama sistem komputer dan program tidak dirubah.
2.Proses transfer dari manusia satu ke lainnya membutuhkan proses yang lama	2.Lebih mudah diduplikasi dan disebarkan.
3.Lebih mahal karena tidak jarang harus mendatangkan orang untuk suatu pekerjaan	3.Menyediakan layanan komputer akan lebih mudah dan murah dibandingkan mendatangkan seseorang untuk mengerjakan sejumlah pekerjaan dalam jangka waktu yang sangat lama.
4.Sering berubah-ubah karena sifat manusia.	4. Konsisten karena kecerdasan buatan merupakan bagian dari sistem komputer.

5.Kecerdasan alami sangat sulit untuk direproduksi	5.Dapat didokumentasi. Keputusan yang dibuat komputer dapat didokumentasi dengan mudah dengan cara melacak setiap aktivitas dari sistem tersebut.
6.Lebih lambat. Mentransfer pengetahuan manusia dari satu orang ke orang lain membutuhkan proses yang sangat lama.	6.Lebih cepat. Jika pengetahuan terletak pada suatu sistem komputer, maka pengetahuan tersebut dapat disalin dari komputer dengan cepat.
7.Seringkali kurang teliti	7.Dapat mengerjakan pekerjaan lebih baik

Manfaat intelektual alami antara lain :

1. Inovatif, keterampilan untuk menambah pengetahuan sangat erat pada diri manusia.
2. Memperbolehkan manusia memanfaatkan pengalaman secara langsung.
3. Pandangan manusia dapat diterapkan secara logis.

## **F. Penerapan Artificial Intelligence**

Mesin cerdas adalah mesin yang mampu bertindak seperti dan sebaik tindakan atau kerja yang dilakukan oleh manusia. Oleh karena itu mesin cerdas perlu ditambahkan bekal intelegensia sehingga memiliki kualifikasi untuk berfikir, dimana dibutuhkan 2 bagian utama yang sangat dibutuhkan meliputi :

1. Basis pengetahuan (*knowledge base*), bersifat kenyataan-kenyataan, teori, pendapat, dan hubungan antara satu dengan yang lainnya,
2. Mesin inferensi (*inference engine*), kemampuan mengambil keputusan berdasar atas kepakaran dan pengetahuan.

Penerapan *artificial intelligence* meliputi :

1. System Pakar. Disini komputer dimanfaatkan sebagai fasilitas untuk mencatat pengetahuan para ahli. Dengan demikian komputer akan mempunyai keahlian untuk menyelesaikan permasalahan dengan mengambil alih ilmu pengetahuan para pakar yang dipunyai oleh seorang pakar.
2. Pemrosesan Bahasa alami. Dengan pemrosesan bahasa alamiah ini diharapkan pengguna mampu berinteraksi dengan komputer dengan menggunakan bahasa sehari-hari.
3. Pengenalan Suara (*Speech Recognition*). Melalui pengenalan suara diharapkan manusia mampu berinteraksi dengan komputer dengan menggunakan suara.
4. Robotika dan Sistem Sensor (*Robotics and Sensor Systems*).
5. Computer Vision, mencoba untuk dapat menginterpretasikan gambar atau obyek-obyek tampak melalui komputer.
6. Instruksi Bantuan Komputer yang cerdas. Komputer dapat digunakan sebagai pemandu yang dapat mengajarkan dan melatih.
7. Permainan yang cerdas.

Seiring dengan perkembangan problem yang dialami manusia, maka *human* perlu memanfaatkan sumber daya dengan baik. Untuk melaksanakan hal tersebut, maka manusia memerlukan bantuan dengan kapasitas yang baik dari komputer. Dalam kehidupan manusia, komputer mampu menolong dalam berbagai hal meliputi :

1. Pertanian, komputer mampu menggerakkan mesin yang melakukan pendeteksian hama, pemotongan pohon, memilih hasil panen.

2. Manufaktur (Industri), komputer mampu mengontrol mesin yang harus melaksanakan pekerjaan beresiko tinggi, pengawasan dan mengerjakan perawatan perusahaan.
3. Kesehatan, komputer mampu melakukan pekerjaan dokter dalam mendiagnosa penyakit, melakukan monitoring keadaan pasien, menghasilkan perawatan yang sesuai.
4. Aktivitas rumah tangga, komputer mampu memberikan peringatan dalam aktivitas yang ada di dalam rumah tangga, hingga mampu membantu membersihkan rumah, menjaga rumah dari kecurian, serta hal-hal yang lain.

## **G. Soft Computing**

*Soft computing* adalah koleksi dari beberapa metode yang bertujuan untuk mengeksploitasi toleransi terhadap ketidaktepatan, ketidakpastian, dan kebenaran parsial untuk dapat diselesaikan dengan mudah, robustness dan biaya penyelesaiannya murah (Zadeh, 1992).

*Soft computing* adalah inovasi baru dalam membangun sistem cerdas. Sistem cerdas ini merupakan sistem yang memiliki keahlian seperti manusia pada domain tertentu, mampu beradaptasi dan belajar agar dapat bekerja lebih baik jika terjadi perubahan lingkungan.

Unsur-unsur pokok dalam *Soft computing* yaitu :

### 1. Fuzzy Logic

*Fuzzy logic* merupakan representasi dari suatu pengetahuan yang dikonstruksikan dengan if-then rules. Karakteristik dari sistem fuzzy logic antara lain pemecahan problem dilakukan dengan menjelaskan sistem bukan

melalui angka, adanya pemakaian if-then rules, menjelaskan sistem menggunakan berbagai algoritma fuzzy.

Kelebihan-kelebihan dari fuzzy logic meliputi :

- Dapat mengekspresikan konsep yang susah untuk dirumuskan, misalnya suhu ruangan yang nyaman.
- Pemakaian membership function memungkinkan fuzzy untuk melakukan observasi obyektif terhadap nilai-nilai yang subyektif.
- Penerapan logika dalam pengambilan keputusan

## 2. Jaringan saraf tiruan (artificial neural network)

Jaringan saraf tiruan adalah prosesor yang terdistribusi paralel, terbuat dari unit-unit yang sederhana, dan memiliki kemampuan untuk menyimpan pengetahuan yang diperoleh secara eksperimental dan siap pakai untuk berbagai tujuan. Jaringan saraf tiruan ini meniru otak manusia dari sudut pengetahuan melalui proses pembelajaran (*training*), kekuatan koneksi antar unit yang disebut synaptic weights berfungsi untuk menyimpan pengetahuan yang telah diperoleh jaringan tersebut.

## 3. Probabilistic reasoning

Reasoning berarti mengambil suatu keputusan atas suatu alasan atau sebab tertentu. Dua jenis reasoning adalah logical reasoning dan probabilistic reasoning. Salah satu kelebihan probabilistic reasoning dibandingkan logical reasoning terletak pada kemampuan untuk mengambil keputusan yang rasional, walaupun informasi yang diolah kurang lengkap atau mengandung unsur ketidakpastian. Termasuk dalam kategori probabilistic reasoning antara lain teori Chaos, Belief Networks, Genetic Algorithm.

#### 4. Evolutionary computing

*Evolutionary computation* merupakan suatu substansi dari *computer science* yang mengaplikasikan paradigma intelektual dari rancangan dan pandangan dasar dari perkembangan alam, adalah pandangan seleksi alam Darwinisme, sebagai gagasan dalam perancangan metode komputasi. Dalam mekanisme penyaringan semesta, siapa yang kuat (yang bisa beradaptasi) dialah yang bisa bertahan. Ternyata pendapat ini telah merebak sejak tahun 1940an, jauh sebelum zaman perkembangan komputer yang semakin pesat. Pada tahun 1948, Turing mempublikasikan istilah “*genetical or evolutionary search*” dan tahun 1962 Bremermann mengerjakan sebuah percobaan mengenai “optimisasi melalui evolusi dan kombinasi ulang (*optimization through evolution and recombination*)”. Pada zaman 1960an, tiga penerapan gagasan utama ini dikembangkan masing-masing di wilayah yang berbeda. Di Amerika, Fogel, Owens, dan Walsh memublikasikan *Evolutionary Programming*, sedangkan Holland (juga di Amerika) mengatakan bahwa prosedurnya sebagai *Genetic Algorithm*. Sementara itu di Jerman, Rechenberg dan Schwefel mendapatkan prosedur *Evolution Strategies*. Selama 15 (limabelas) tahun berikutnya, prosedur tersebut dikembangkan secara terpisah, namun sejak awal tahun 1990-an ketiganya dipandang sebagai tiga jenis perwakilan (dialek) dari sebuah teknologi yang dinamai *Evolutionary Computing*. Di permulaan zaman 1990an juga bergabung dalam perkembangan pemikiran ini suatu prosedur yang baru, yaitu *Genetic Programming*, yang dipelopori oleh Koza.

Karakteristik *Soft computing* meliputi :

1. *Soft computing* membutuhkan kepakaran human, apabila direpresntasikan dalam susunan kaidah If-then
2. Paradigma komputasinya didasari oleh tahapan biologis
3. *Soft computing* menerapkan proses komputasi angka
4. *Soft computing* mempunyai toleransi kegagalan, meskipun kualitasnya berlahan-lahan menurun.

#### **H. Latihan 1**

1. Jelaskan evolusi dan perkembangan Artificial Intelligence !
2. Apakah Artificial Intelligence itu ? jelaskan pula definisi Artificial Intelligence dari 4 perspektif (sudut pandang )
3. Jelaskan 9 tujuan dari adanya kecerdasan buatan disertai dengan contohnya!
4. Jelaskan Perbedaan Kecerdasan Alami dan Kecerdasan Buatan !
5. Apa yang anda ketahui tentang *soft computing* dan jelaskan karakteristik dari *soft computing* ?

# BAB 2

## Algoritma Pencarian (*Searching Algorithm*)

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi algoritma penelusuran
  - ❖ Menjelaskan struktur permasalahan algoritma penelusuran
  - ❖ Menjelaskan jenis algoritma penelusuran
- 

### A. Definisi Algoritma Pencarian

Pencarian adalah sesuatu aktifitas mencari penyelesaian dari sesuatu persoalan melalui sekumpulan probabilitas keadaan ruang (*state of space*). Keadaan Ruang merupakan sesuatu ruang yang berisi semua keadaan yang mungkin. Di dalam ilmu komputer, sebarang algoritma penelusuran diturunkan secara lebar merupakan algoritma yang mengizinkan inputan berupa sebuah problem dan mengeluarkan sesuatu solusi untuk problem tersebut, yang biasanya didapat dari evaluasi beberapa kemungkinan solusi. Kumpulan semua kemungkinan penyelesaian dari sebuah problem disebut ruang penelusuran. Algoritma penelusuran *brute-force* atau penelusuran uninformed menggunakan metode yang sederhana dan sangat intuitif pada ruang penelusuran, sedangkan algoritma penelusuran informed menggunakan heuristik untuk menerapkan pengetahuan tentang struktur dari ruang penelusuran untuk berusaha mengurangi banyaknya waktu yang dipakai dalam penelusuran (Nilson. J, 1998).

Algoritma penelusuran adalah metode penelusuran terhadap ruang permasalahan, bukan pada data. Proses penelusuran berupa jalur yang menggambarkan keadaan awal sebuah problem menuju pemecahan. Proses penelusuran menuju pemecahan akan membentuk sebuah *solution space*.

Contoh pengaplikasian algoritma penelusuran dapat dilihat pada penentuan rute, navigasi robot, perencanaan pada block wordl, eight-queen problem pada catur, eight puzzle, dan berbagai macam contoh lainnya.

Untuk memperkirakan performa metode penelusuran, terdapat empat standarisasi yang dapat digunakan (Coppin. B, 2004) :

1. *Completeness* : apakah metode tersebut menjamin ditemukannya penyelesaian apabila penyelesaiannya memang ada,
2. *Time complexity* : berapa lama waktu yang diperlukan,
3. *Space complexity* : berapa banyak memori yang diperlukan,
4. *Optimality* : apakah metode tersebut menjamin ditemukannya penyelesaian yang terbaik apabila terdapat beberapa penyelesaian berbeda.

## **B. Struktur Perproblem Algoritma Penelusuran**

Struktur perproblem pada algoritma *penelusuran* diantaranya :

1. Berhadapan dengan situasi awal dan ada *goal* tertentu yang hendak dicapai.
2. Pada kondisi tertentu, ada pilihan dari tindakan sederhana untuk dipilih, contohnya belok ke kiri atau belok ke kanan. Melakukan urutan tindakan tertentu dapat membuat *goal* tercapai atau juga tidak tercapai. Oleh karena itu, urutan harus terdefinisi dengan jelas.

3. Penelusuran adalah proses meninjau beberapa urutan langkah dan memilih salah satu yang akan mencapai *goal*.
4. Untuk beberapa aplikasi, setiap urutan tindakan terkait dengan nilai tertentu. Perproblememan penelusuran tidak hanya untuk mencapai goal, tetapi juga nilai maksimum dalam mencapai goal tersebut.

### C. Jenis Algoritma Penelusuran

Algoritma penelusuran menurut jenisnya secara umum ada 2 diantaranya :

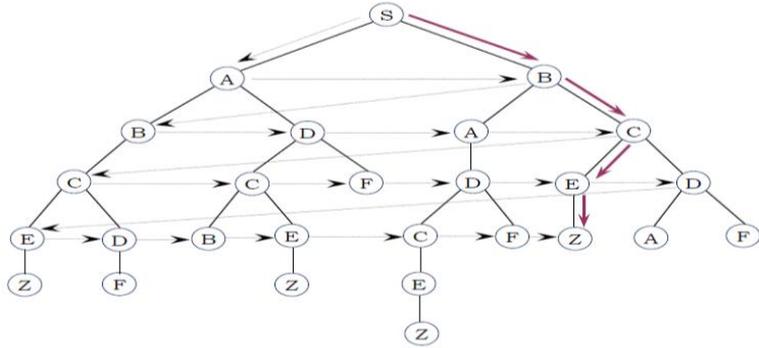
1. Penelusuran Buta (*Blind Search* atau *Uninformed search algorithm*)

Algoritma yang tidak memberikan informasi atau pengetahuan mengenai perproblememan yang ada. Isinya hanya sebatas definisi dari algoritma tersebut.

Algoritma ini meliputi :

- a. Breadth first

Pada metode breadth first, semua node pada level  $n$  dikunjungi terlebih dahulu sebelum mengunjungi node-node pada level  $n+1$ . Penelusuran dimulai dari node akar terus ke level ke-1 dari kiri ke kanan, kemudian berpindah ke level berikutnya demikian pula dari kiri ke kanan sampai ditemukannya penyelesaian.



### Algoritma Breadth first

- Buat sebuah antrian, inialisasi node pertama dengan root dari tree
- Bila node pertama, apabila  $\neq$  goal, diganti dengan anak-anaknya dan diletakkan di belakang per level
- Bila node pertama = goal, selesai

### Keuntungan :

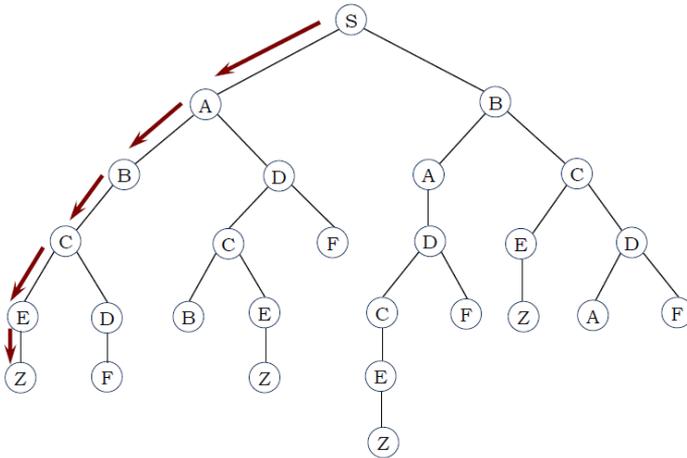
- Tidak akan menemui jalan buntu
- Apabila ada satu penyelesaian, maka breadth search akan menemukannya. Dan apabila ada lebih dari satu penyelesaian maka penyelesaian minimum akan ditemukan.

### Kelemahan :

- Membutuhkan memori yang cukup banyak, karena menyimpan semua node dalam satu pohon
- Kemungkinan ditemukan optimal lokal

b. Depth first Search

Depth first search, merupakan mekanisme penelusuran akan dilaksanakan pada semua turunannya sebelum dilakukan penelusuran ke node-node yang setingkat. Penelusuran diawali dari node akar ke tingkat yang lebih tinggi. Mekanisme ini diulangi terus hingga ditemukannya penyelesaian.



Algoritma depth first search

- Membuat sesuatu antrean, inialisasi node pertama dengan root dari tree
- Apabila node pertama, apabila  $\neq$  sasar, node dihapus diganti dengan turunan-turunannya dengan urutan LChild
- Apabila node pertama = goal, selesai

Keuntungan :

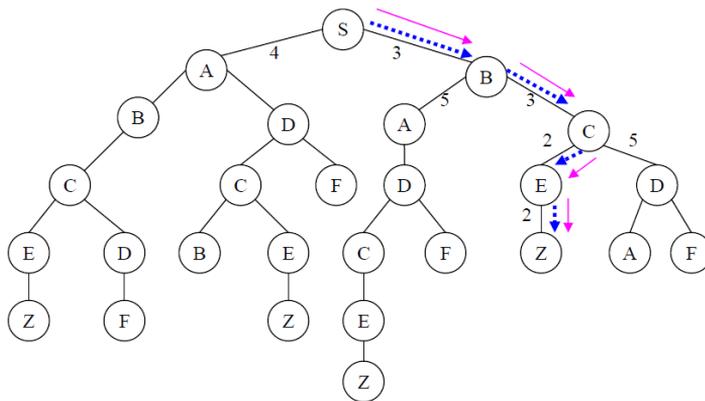
- Memerlukan penyimpanan yang relatif kecil, dikarenakan hanya node-node pada lintasan yang aktif saja yang disimpan.

- Menemukan penyelesaian tanpa perlu menguji lebih banyak lagi dalam ruang keadaan.

Kelemahan :

- Kemungkinan terjebak pada optimal lokal
- Hanya akan mendapatkan 1 penyelesaian pada setiap penelusuran

### c. Hill Climbing



Algoritma Hill climbing search

- Membuat sebuah antrean, inialisasi node pertama dengan root dari tree
- Apabila node pertama, apabila  $\neq$  goal, node dihapus diganti dengan turunan-turunannya dengan urutan yang paling kecil jaraknya
- Apabila node pertama = goal, selesai

Keuntungan Hill climbing search:

- Memerlukan penyimpanan yang cukup kecil, dikarenakan hanya node-node pada lintasan yang aktif saja yang disimpan.

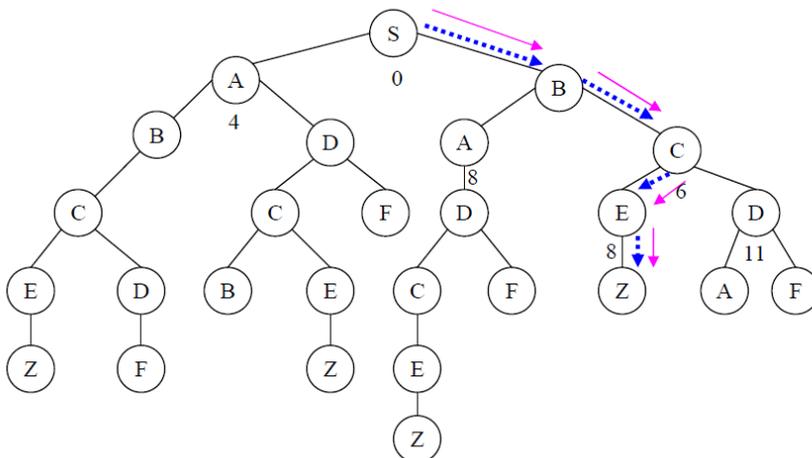
- Prosedur *hill climbing search* akan menemukan penyelesaian tanpa harus menguji lebih banyak lagi dalam ruang keadaan.

Kelemahan Hill climbing search:

- Algoritma akan berhenti kalau mencapai nilai optimum lokal
- Perlu menentukan aturan yang tepat.

d. Best first search

Prosedur *best first search* ini merupakan kombinasi dari beberapa kelebihan teknik *depth first search* dan *breadth first search*. Pada penelusuran dengan *hill climbing* tidak diperkenankan untuk kembali ke node pada level yang lebih rendah meskipun node pada level yang lebih rendah tersebut memiliki nilai heuristik yang lebih baik. Hal ini berbeda dengan prosedur *best first search* dimana penelusuran diperkenankan mengunjungi node yang ada di level yang lebih rendah apabila ternyata node pada level yang lebih tinggi ternyata memiliki heuristik yang buruk.



### Algoritma Best first search

- Buat sebuah antrean, inialisasi node pertama dengan root dari tree
- Bila node pertama, apabila  $\neq$  goal, node dihapus diganti dengan anak-anaknya. Selanjutnya keseluruhan node yang ada di Queue di-sort ascending
- Bila node pertama = goal, selesai

### Keuntungan Best first search:

- Memerlukan penyimpanan yang relatif kecil, karena hanya node-node pada lintasan yang aktif saja yang disimpan
- Prosedur *Best first search* akan menemukan penyelesaian tanpa harus menguji lebih banyak lagi dalam ruang keadaan.

### Kelemahan Best first search:

- Algoritma akan berhenti kalau mencapai nilai optimum lokal
- Tidak diijinkan untuk melihat satupun langkah sebelumnya

## 2. Penelusuran Optimal

Pada penelusuran optimal dikenal beberapa algoritma yaitu :

### a. Branch and bound

Pada prosedur ini tidak memerlukan perkiraan jarak tiap node menuju goal, selain itu elemen-elemen pada Queue bukan tiap node, melainkan lintasan parsial yang sudah tercapai.

### Algoritma Branch and bound

- Buat sebuah antrean, inialisasi node pertama dengan root dari tree

- Bila lintasan parsial  $\neq$  lintasan goal, maka lintasan parsial diganti dengan lintasan parsial + node child, semuanya diatur berdasarkan harga yang diurut secara ascending.
- Bila node pertama = lintasan goal, selesai.

Keuntungan Branch and bound:

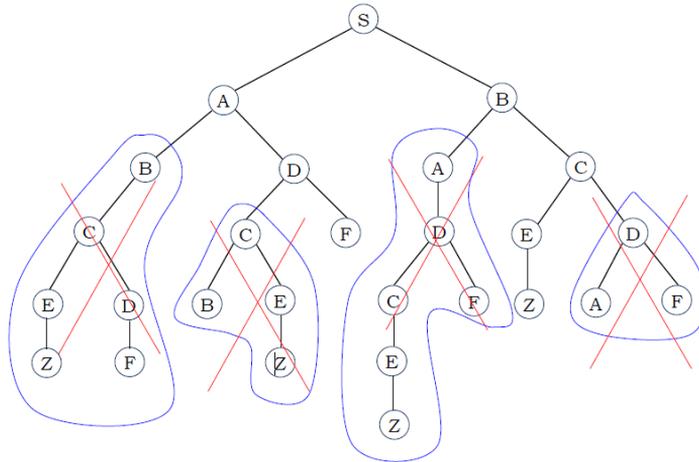
- Algoritma berhenti pada nilai optimum sebenarnya (menentukan optimum global)

Kelemahan Branch and bound:

- Memerlukan penyimpanan yang cukup banyak, karena bisa jadi menyimpan semua lintasan parsial yang memungkinkan

b. Branch and bound dengan Pemrograman Dinamis

Prosedur ini sama dengan *Branch and bound*, tetapi lebih efisien karena bisa mengurangi lebar / melakukan pemotongan terhadap lebar dari tree. Hal ini dilakukan dengan cara mereduksi lintasan parsial yang menuju ke suatu node yang sudah pernah dikunjungi sebelumnya.



### Algoritma Branch and bound dengan Pemrograman Dinamis

- Buat sebuah antrian, inialisasi node pertama dengan root dari tree
- Bila lintasan parsial  $\neq$  lintasan goal, apabila ada lintasan parsial dengan node terakhir yang sama (dalam satu queue) maka diambil yang harganya paling minimal, sedangkan yang lebih mahal dihapus dengan queue, sehingga tree akan lebih kurus.
- Bila node pertama = lintasan sasaran, selesai.

### Keuntungan *Branch and bound* dengan Pemrograman Dinamis:

- Algoritma berhenti pada nilai optimum sebenarnya
- Lebih efisien dari prosedur *Branch and bound* dalam penggunaan penyimpanan dan waktu eksekusi karena ada pemotongan

### Kelemahan Branch and bound:

- Harus mengingat node terakhir dari lintasan parsial yang sudah dicapai sebelumnya.

### 3. Penelusuran Untuk Game

Permainan adalah sesuatu yang sangat menarik dan menjadi sub topik tersendiri di dalam kecerdasan buatan. Terdapat beberapa alasan mengapa permainan menjadi menarik yaitu :

- a. Standarisasi menang atau kalah jelas
- b. Dapat mempelajari perproblem
- c. Alasan historis
- d. Menyenangkan
- e. Biasanya mempunyai search space yang besar

Beberapa ciri umum pada permainan (game) dalam kecerdasan buatan yaitu :

- Terdapat 2 pemain
- Kesempatan pemain bergantian
- Zero sum, kerugian seorang pemain adalah keuntungan untuk pemain lain
- Perfect information, pemain mengetahui semua informasi state dari game
- Tidak mengandung probabilitas (seperti dadu)
- Contoh : tic tac toe, checkers, chess, go, nim dan othello
- Game tidak termasuk bridge, solitaire, backgammon, dan semisalnya

### D. Contoh Kasus

Terdapat 2 buah tabung masing-masing mempunyai volume 5 liter yaitu tabung A dan 2 liter untuk tabung B. Tidak mempunyai tanda yang menunjukkan batas ukuran pada kedua tabung tersebut. Ada sebuah pompa air akan digunakan untuk

mengisi air pada kedua tabung tersebut. Problemnya bagaimana dapat mengisi air tepat 5 liter ke dalam tabung yang berkapasitas 5 liter pada tabung A tersebut ?

Jawaban :

1. Langkah Pertama : Identifikasi Ruang Keadaan

Perprobleman ini dapat diselesaikan dengan 2 bilangan integer, yaitu X dan Y dimana :

X = air yang diisikan pada tabung 5 liter (tabung A)

Y = air yang diisikan pada tabung 2 liter (tabung B)

Ruang keadaan (X,Y) sedemikian hingga  $X \in \{0,1,2,3,4\}$  dan  $Y \in \{0,1,2,3\}$

2. Langkah Kedua : Keadaan awal dan tujuan

Keadaan awal, kedua tabung dalam keadaan kosong (0,0), Tujuannya, pada tabung A berisi tepat 5 liter air (5,n) untuk sembarang n.

3. Langkah Ketiga : Rule (aturan) yang digunakan

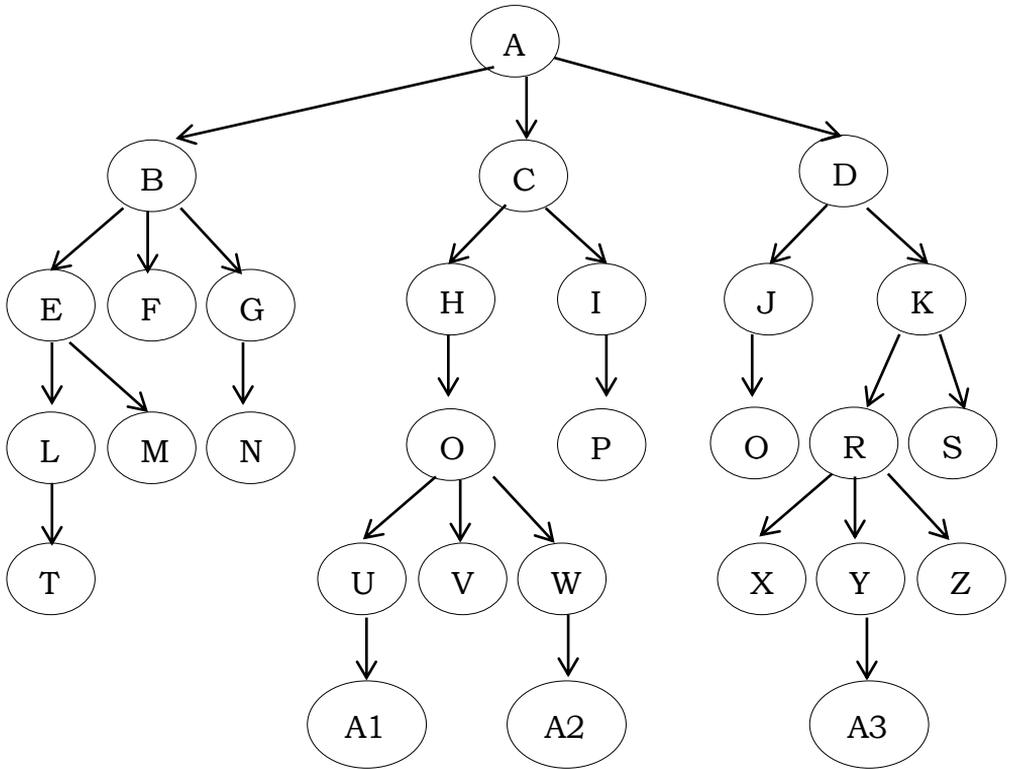
- Aturan 1 : If  $X < 5$  then (5,Y)  
isi tabung A
- Aturan 2 : If  $Y < 2$  then (X,2)  
isi tabung B
- Aturan 3 : If  $X > 0$  then (X-d,Y)  
Keluarkan sebagian air keluar dari tabung A.
- Aturan 4 : If  $Y > 0$  then (X,Y-d)  
Tuangkan sebagian air keluar dari tabung B.
- Aturan 5 : If  $X > 0$  then (0,Y)  
Kosongkan tabung A dengan membuang airnya ke tanah.
- Aturan 6 : If  $Y > 0$  then (X,0)

Kosongkan tabung B dengan membuang airnya ke tanah.

- Aturan 7 : If  $X + Y \geq 2$  dan  $Y > 0$  then  $(5, Y - \{5 - X\})$   
 Tuang air dari tabung B ke tabung A sampai tabung A penuh
- Aturan 8 : If  $X + Y \geq 2$  dan  $X > 0$  then  $(X - \{2 - Y\}, Y)$   
 Tuang air dari tabung A ke tabung B sampai tabung B penuh
- Aturan 9 : If  $X + Y \leq 5$  dan  $Y > 0$  then  $(X + Y, 0)$   
 Tuang air dari tabung B ke tabung A
- Aturan 10 : If  $X + Y \leq 5$  dan  $X > 0$  then  $(0, X + Y)$   
 Tuang air dari tabung A ke tabung B
- Aturan 11 : If  $(0, 5)$  then  $(5, 0)$   
 Tuang 2 liter air dari tabung B ke tabung A
- Aturan 12 : If  $(5, Y)$  then  $(0, Y)$   
 Kosongkan 2 liter air di tabung A dengan membuang airnya ke tanah.

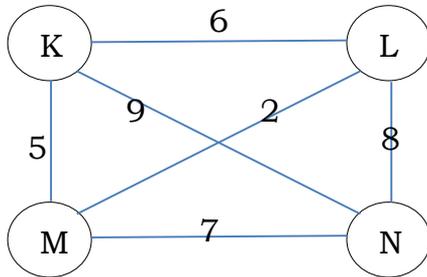
## I. Latihan 2

1. Jelaskan apa yang ada ketahui tentang algoritma penelusuran ?
2. Sebut dan jelaskan 4 jenis Penelusuran Buta (*Blind Search* atau *Uninformed search algorithm*) !
3. Sebut dan jelaskan 2 jenis penelusuran optimal !
4. Diketahui representasi ruang keadaan sebagai berikut :



Bila keadaan awal berada di node A dan diinginkan ada 3 tujuan, yaitu node M, node A1 dan node Z, implementasikan algoritma penelusuran BFS (*Bread First Search*) dan DFS (*Depth First Search*) untuk pohon keadaan diatas untuk mencari penyelesaiannya, berupa lintasan dari node awal sampai dengan node tujuan.

5. Seorang salesman ingin mengunjungi 4 kota yaitu K, L, M, dan N seperti gambar berikut :



Jarak terpendek tiap-tiap kota sudah diketahui. Salesman tersebut ingin mencari rute terpendek dimana setiap kota boleh dikunjungi tepat 1 kali dan ia boleh memilih keadaan awalnya di suatu kota dan keadaan akhirnya di kota yang lain. Selesaikan persoalan TPS tersebut dengan algoritma hill climbing!

6. Misalkan Anda mempunyai 2 buah ember, masing-masing bervolume 7 liter dan 5 liter. Anda diminta mendapatkan air (dari sebuah danau) sebanyak 4 liter di dalam ember yang bervolume 7 liter dan 5 liter bervolume 5 kuter dengan menggunakan bantuan hanya kedua ember tersebut (tidak ada peralatan lain yang tersedia, hanya kedua ember itu saja yang ada). Anda boleh memindahkan air dari satu ember ke ember lain, membuang seluruh air dari ember, dan sebagainya. Gambarkan penelusuran penyelesaian persoalan ini dengan membangun pohon secara dinamis dengan algoritma BFS (*Bread First Search*) dan DFS (*Depth First Search*)!

# BAB 3

## Representasi Pengetahuan

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang representasi pengetahuan
  - ❖ Menjelaskan tentang peran representasi pengetahuan
  - ❖ Menjelaskan tentang klasifikasi representasi pengetahuan
  - ❖ Menjelaskan tentang syarat representasi pengetahuan
  - ❖ Menjelaskan tentang keuntungan representasi pengetahuan
  - ❖ Menjelaskan tentang logika sebagai representasi pengetahuan
- 

### A. Definisi Representasi Pengetahuan

Dalam menyelesaikan problem tentunya memerlukan pengetahuan yang mencukupi. Pengetahuan adalah kenyataan atau kondisi keadaan yang hadir karena sesuatu pengalaman. Pengetahuan adalah sesuatu yang muncul dan terbentuk dalam jiwa dan pikiran seseorang karena adanya reaksi, sentuhan dan hubungan dengan lingkungan dan alam sekitarnya. Contoh: pengetahuan tentang binatang, sifat dan perilaku binatang; pengetahuan tentang penyakit, gejala dan pengobatannya; pengetahuan tentang tanaman, jenis dan cara hidupnya.

Representasi ditujukan untuk menemukan sifat-sifat penting problema dan membuat informasi tersebut dapat diakses oleh prosedur pemecahan problematika. Bahasa representasi harus mampu membuat seseorang ahli bahasa program mampu mengekspresikan pengetahuan yang dibutuhkan untuk mendapatkan penyelesaian problematika. *Knowledge* (pengetahuan) adalah penjelasan mengenai ranah ilmu yang mampu dimanfaatkan untuk menyelesaikan problematika dalam

ranah ilmu tersebut. Untuk menyelesaikan banyak problematika, diperlukan banyak keahlian. Keahlian tersebut harus diterapkan ke dalam komputer. Sebagai bagian perancangan bahasa pemrograman dalam merampungkan sebuah problem, definisi pengetahuan harus diterapkan dengan jelas.

Representasi pengetahuan dan kemampuan untuk melakukan penalaran dalam bidang kecerdasan buatan merupakan hal penting. Meskipun suatu sistem memiliki banyak pengetahuan, namun apabila tidak memiliki kemampuan untuk menalar akan percuma saja. Sebaliknya apabila sistem memiliki kemampuan yang handal untuk menalar, tetapi basis pengetahuan yang dimilikinya tidak mencukupi, maka penyelesaian yang diperolehnya menjadi tidak maksimal.

Representasi pengetahuan merupakan suatu cara dalam menyaapabilan berbagai informasi (pengetahuan) yang diperoleh ke dalam suatu diagram sehingga diketahui relasi antara pengetahuan yang satu dengan pengetahuan yang lain dan dapat digunakan dalam pengujian logika kebenarannya.

Representasi pengetahuan digunakan untuk menangkap sifat penting problem dan mempermudah prosedur pemecahan problem dalam mengakses informasi. Format representasi pengetahuan harus mudah dipahami sehingga seorang programmer mampu mengekspresikan pengetahuan (fakta).

Banyak cara untuk menyatakan representasi pengetahuan dan setiap cara mempunyai kelebihan dan kelemahan masing-masing. Adapun cara untuk representasi pengetahuan adalah :

- Kalkulus predikat
- Logika
- Pohon
- Frame

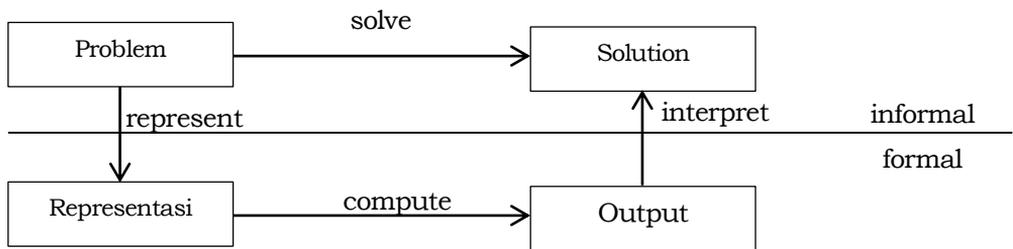
- Script
- Semantic network

Dalam representasi pengetahuan ada dua hal yang harus diperhatikan yaitu :

1. Fakta, dapat berupa keterangan / kalimat dalam bahasa alami, logika atau benda.
2. Formula, bentuk representasi yang dapat dimanipulasi dalam mekanisme pemecahan soal. Bentuk ini harus melukiskan hubungan antara komponen-komponen domain problem.

### B. Peran Representasi Pengetahuan

Representasi dari beberapa potongan pengetahuan adalah representasi internal dari pengetahuan. Secara mendasar, problem dalam konteks umum dapat dibuat penyelesaiannya dengan intuisi atau kemampuan logika user seperti gambar berikut:



Secara mendasar, problem dalam konteks umum dapat dibuat penyelesaiannya dengan intuisi atau kemampuan logika user seperti gambar diatas. Secara formal, ketika dari perproblem diolah menjadi knowledge dalam bentuk representasi pengetahuan, maka akan melibatkan komputasi dalam menentukan output. Output

yang diperoleh dari kalkulasi representasi pengetahuan, akan diinterpretasikan ke dalam penyelesaian permasalahannya.

Sehingga bisa diambil kesimpulan bahwa ketika kita mampu membuat representasi pengetahuan dengan sempurna, maka representasi tersebut mampu dijadikan basis pengetahuan yang baik dalam memecahkan masalah atau memberikan pelayanan yang tepat dalam mengambil kesimpulan.

### **C. Klasifikasi Representasi Pengetahuan**

Representasi pengetahuan dapat diklasifikasikan menjadi 4 kategori antara lain :

1. Representasi logika : representasi jenis ini menggunakan ekspresi-ekspresi dalam logika formal untuk merepresentasikan basis pengetahuan.
2. Representasi prosedural : representasi ini menggambarkan pengetahuan sebagai kumpulan instruksi untuk memecahkan suatu problema.
3. Representasi network : representasi ini menangkap pengetahuan sebagai sebuah graf dimana simpul-simpulnya menggambarkan obyek atau konsep dari problema yang dihadapi, sedangkan edge-nya menggambarkan hubungan atau asosiasi antar mereka.
4. Representasi terstruktur : representasi terstruktur memperluas network dengan cara membuat setiap simpulnya menjadi sebuah struktur data kompleks.

### **D. Syarat Representasi Pengetahuan**

Pengetahuan harus direpresentasikan dengan baik supaya kesimpulan yang diambil tepat. Sebuah representasi yang baik memiliki syarat seperti berikut :

1. Cukup kaya untuk mengekspresikan pengetahuan yang dibutuhkan untuk menyelesaikan permasalahan.

2. Sedekat mungkin dengan permasalahannya. Representasi harus padat, alami dan dapat di-mantain. Representasi juga harus memudahkan melihat hubungan diantara representasi dengan domain yang direpresentasikan. Alasanya, supaya mudah dalam menentukan ketepatan representasi pengetahuan.
3. Dapat digunakan untuk sebuah komputasi yang efisien. Artinya, representasi harus mampu untuk mengekspresikan fitur dari masalahnya.
4. Dapat diperoleh dari orang, data dan pengalaman di masa lalu.

### **E. Keuntungan Representasi Pengetahuan**

Keuntungan yang dapat diperoleh dari representasi pengetahuan meliputi :

1. Dengan representasi yang baik, membuat objek dan relasi yang penting menjadi jelas
2. Representasi menyingkap batasan (*constrain*) dalam suatu permasalahan. Kita dapat mengungkapkan pengaruh sebuah objek atau relasi terhadap objek atau relasi yang lain.
3. Dengan representasi kita akan dapatkan objek dan relasi secara bersama-sama.
4. Kita dapat menghilangkan semua komponen yang tidak berhubungan dengan permasalahan yang sedang kita selesaikan. Atau kita dapat menyembunyikan beberapa informasi yang tidak kita butuhkan untuk sementara, dan pada saat kita memerlukannya kita dapat menampilkan kembali.
5. Dengan representasi akan membuat permasalahan yang sedang kita selesaikan menjadi transparan. Kita akan memahami permasalahan yang kita selesaikan.
6. Dengan representasi kita akan dapat menyingkap suatu permasalahan secara lengkap, sehingga permasalahan dapat terselesaikan

7. Dengan representasi akan membuat perprobleman menjadi ringkas. Kita akan berpikir ringkas (merepresentasikan apa yang ingin kita representasikan secara efisien)
8. Dengan representasi, maka akan menjadikan pekerjaan kita menjadi cepat
9. Dengan representasi, menjadikan perprobleman yang kita selesaikan dapat terkomputerisasi sehingga kita akan dapat melakukan prosedur-prosedur dalam menyelesaikan suatu perprobleman.

### **F. Logika Sebagai Representasi Pengetahuan**

Logika adalah bentuk representasi pengetahuan yang paling tua. Representasi jenis ini menggunakan ekspresi-ekspresi dalam logika formal untuk merepresentasikan basis pengetahuan. Pada dasarnya mekanisme logika adalah mekanisme membentuk kesimpulan dan menarik suatu inferensi berdasarkan fakta yang telah ada. Input dari mekanisme logika berupa premis dan fakta-fakta yang diakui kebenarannya sehingga dengan melakukan penalaran pada mekanisme logika dapat dibentuk suatu inferensi atau kesimpulan yang benar pula.

Contoh fakta sederhana yang direpresentasikan secara logika :

Helder adalah anjing

Fakta dalam bahasa Inggris tersebut direpresentasikan secara logika yaitu :

Anjing (helder)

Kita juga dapat merepresentasikan secara logika fakta lain, yaitu bahwa semua anjing mempunyai ekor :

$\forall x : \text{Anjing}(x) \rightarrow \text{berekor}(x)$

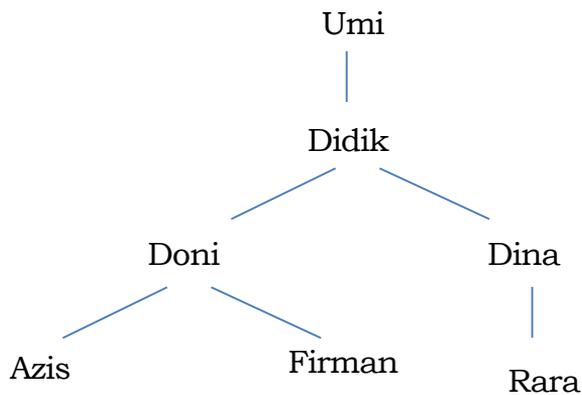
Kemudian secara deduktif (penalaran dimulai dari prinsip umum untuk mendapatkan konklusi yang lebih khusus) dari mekanisme logika ini kita bisa mendapatkan representasi baru :

Berekor (anjing)

Dengan menggunakan fungsi mapping secara backward, kita dapat men-generate kalimat bahasa inggris

Helder berekor

Representasi pengetahuan secara logika untuk tree dapat dilihat seperti pada gambar berikut :



Dari tree diatas kita dapat membuat representasinya secara logika sebagai berikut :

Putra (Umi, Didik)

Putra (Didik, Doni)

Putra (Doni, Azis)

Putra (Doni, Firman)

Putri (Didik, Dina)

Putri (Dina, Rara)

Seperti diungkapkan diatas dengan representasi secara logika kita dapat memberikan pertanyaan dari fakta-fakta yang sudah direpresentasikan. Bentuk pertanyaan siapa putri dari Dina ? maka untuk representasi secara logikanya dapat kita menyatakannya sebagai berikut :

Putri (Dina, D)

D = Rara

D adalah sebuah variabel yang kita gunakan untuk menampung jawaban dari pertanyaan siapa putri dari Dina. Dari pertanyaan yang dipresentasikan secara logika putri (Dina, D) kita dapat jawabannya adalah D = Rara.

Bentuk pertanyaan lain yang dapat direpresentasikan secara logika adalah seperti dibawah ini. Bentuk pertanyaannya adalah siapa saja dari struktur keluarga diatas yang mempunyai putri dan siapa nama putrinya. P merupakan variabel yang menampung pertanyaan siapa yang mempunyai putri, sementara D merupakan variabel yang menampung nama putrinya.

Putri(P, D)

P = Didik                      D = Dina

P = Dina                        D = Rara

Apabila kita menanyakan “Apakah Doni mempunyai saudara perempuan atau tidak ? dan apabila mempunyai, siapa namanya ? maka Query dengan representasi logika adalah sebagai berikut :

Putra(P, Doni), putri(P,D)

P = Didik          D = Dina

P adalah variabel yang menampung semua orangtua dari Doni, sementara D adalah variabel yang menampung nama saudara perempuan Doni, sehingga didapatkan jawaban P = Didik dan D = Dina

### **G. Latihan 3**

1. Jelaskan apa yang dimaksud dengan representasi pengetahuan ?
2. Sebut dan jelaskan klasifikasi representasi pengetahuan !
3. Pengetahuan harus direpresentasikan dengan baik supaya kesimpulan yang diambil tepat. Sebutkan beberapa syarat dari representasi pengetahuan yang baik !
4. Apasajakah keuntungan dari representasi pengetahuan ?
5. Dengan representasi secara logika dibawah ini jawablah pertanyaan dibawah ini :

besar (kuda).

besar (sapi).

besar (gajah).

kecil (kelinci).

kecil (tikus).

coklat (kuda).

putih (sapi).

abuabu (gajah).

abuabu (tikus).

putih (kelinci).

Terang (X) : -putih (X)

Terang (X) : -abuabu (X)

Jelaskan hasil dari query berikut ini :

Putih(Y), besar(Y)

abuabu(Y), kecil (Y)

# BAB 4

## Sistem Pakar

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi sistem pakar
  - ❖ Menjelaskan tentang kelebihan dan manfaat sistem pakar
  - ❖ Menjelaskan tentang kekurangan sistem pakar
  - ❖ Menjelaskan tentang ciri-ciri sistem pakar
  - ❖ Menjelaskan tentang ruang lingkup problem dalam sistem pakar
  - ❖ Menjelaskan tentang perbandingan sistem pakar dan sistem konvensional
  - ❖ Menjelaskan tentang membangun sistem pakar
  - ❖ Menjelaskan tentang struktur sistem pakar
  - ❖ Menjelaskan tentang prosedur forward chaining dan backward chaining
- 

### A. Definisi Sistem Pakar

Pakar adalah orang yang memiliki pengetahuan khusus, pendapat, pengalaman, prosedur, serta kemampuan untuk mengaplikasikan keahliannya tersebut guna menyelesaikan problem.

Sistem pakar merupakan salah satu cabang dari kecerdasan buatan yang dikembangkan pada tahun 1960-an. Sistem pakar adalah suatu sistem yang dirancang untuk dapat menirukan keahlian seorang pakar dalam menjawab pertanyaan dan memecahkan suatu problem. Sistem pakar adalah sistem yang berusaha mengadopsi pengetahuan manusia ke komputer agar komputer dapat menyelesaikan problem seperti yang biasa dilakukan oleh manusia yang ahli (pakar). Sistem pakar yang baik dirancang agar dapat menyelesaikan suatu permasalahan tertentu dengan meniru kerja dari para pakar atau ahli. Sistem pakar

akan memberikan pemecahan problem yang didapat dari dialog dengan pengguna. Dengan bantuan sistem pakar, seseorang yang bukan pakar atau ahli dapat menjawab pertanyaan, menyelesaikan problem serta mengambil keputusan yang biasanya dilakukan oleh seorang pakar.

Menurut Efraim Turban (2001), sistem pakar adalah sebuah sistem yang menggunakan pengetahuan manusia dimana pengetahuan tersebut dimasukkan ke dalam sebuah komputer dan kemudian digunakan untuk menyelesaikan problem-problem yang biasanya memerlukan kepakaran atau keahlian manusia.

Menurut Jackson (1999) sistem pakar adalah program komputer yang melakukan dan menyaapabilan penalaran atau logika dengan pengetahuan beberapa pakar untuk memecahkan problem dengan pemberian saran atau rekomendasi.

Menurut Luger dan Stubblefield (1993) sistem pakar adalah program yang berbasiskan pengetahuan yang menyediakan kualitas pakar terhadap problem-problem dalam bidang atau domain yang spesifik.

Sistem pakar adalah program komputer yang menyimulasi penilaian dan perilaku manusia atau organisasi yang memiliki pengetahuan dan pengalaman ahli dalam bidang tertentu. Biasanya, sistem seperti ini berisi basis pengetahuan yang berisi akumulasi pengalaman dan satu set aturan untuk menerapkan pengetahuan dasar untuk setiap situasi tertentu. Sistem pakar yang canggih dapat ditingkatkan dengan penambahan basis pengetahuan atau set aturan. Diantara banyak sistem pakar yang ada, yang terkenal adalah aplikasi bermain catur dan sistem diagnosis medis.

Pemrosesan yang dilakukan oleh sistem pakar merupakan pemrosesan pengetahuan, bukan pemrosesan data pada sistem komputer konvensional, bukan

pemrosesan data pada sistem komputer konvensional. Pengetahuan (knowledge) adalah pemahaman secara praktis maupun teoritis terhadap suatu obyek atau domain tertentu. Pengetahuan yang digunakan pada sistem pakar merupakan serangkaian informasi mengenai gejala diagnosa, sebab akibat, aksi reaksi tentang suatu domain tertentu (misalnya domain diagnosa medis). Beberapa model representasi pengetahuan yang penting adalah :

1. Jaringan semantic (*semantic nets*)
2. Bingkai (*frame*)
3. Kaidah produksi (*production rule*)
4. Logika predikat (*predicate logic*)

Dengan sistem pakar, perproblemaman yang seharusnya hanya dapat diselesaikan oleh para pakar/ahli, sistem pakar membantu aktivitas mereka sebagai asisten yang seolah-olah sudah mempunyai banyak pengalaman.

## **B. Kelebihan dan Manfaat Sistem Pakar**

Beberapa kelebihan, manfaat dan peranan sistem pakar yang populer antara lain :

1. Sistem pakar mampu meningkatkan produktivitas artinya sistem pakar dapat bekerja lebih cepat, efektif dan efisien dibandingkan tenaga manusia. Sistem pakar dapat menghemat waktu dalam pengambilan keputusan.
2. Sistem pakar dapat membuat user yang awam menjadi seperti pakar.
3. Sistem pakar mampu meningkatkan kualitas, dengan memberi nasehat yang konsisten dan mengurangi kesalahan
4. Sistem pakar mampu menangkap pengetahuan dan kepakaran user
5. Sistem pakar memudahkan akses pengetahuan seorang pakar

6. Sistem pakar tergolong handal dalam menyelesaikan banyak perproblememan yang diinputkan
7. Sistem pakar mampu meningkatkan kemampuan sistem komputer sehingga menjadi lebih efektif
8. Sistem pakar mampu bekerja dengan informasi yang tidak lengkap atau tidak pasti.
9. Sistem pakar dapat memberikan informasi yang lengkap dalam prosedur training.
10. Sistem pakar meningkatkan kemampuan dalam pemecahan problem berdasarkan pengetahuan dari pakar.

### **C. Kekurangan Sistem Pakar**

Beberapa kekurangan sistem pakar yaitu:

1. Biaya yang sangat mahal untuk membuat dan maintenance sistemnya
2. Sistem pakar susah dikembangkan karena keterbatasan keahlian pakar
3. Kebenaran hasil uji sistem pakar tidak 100% karena seseorang yang terlibat dalam pembuatan sistem pakar tidak selalu benar. Oleh karena itu perlu diuji ulang secara teliti sebelum digunakan.

### **D. Ciri-ciri Sistem Pakar**

Ciri-ciri sistem pakar meliputi :

1. Terbatas pada domain keahlian tertentu
2. Dapat memberikan alasan-alasan yang tidak lengkap atau tidak pasti
3. Dpaat menjelaskan alasan-alasan dengan cara yang dapat dipahami
4. Dijalankan berdasarkan aturan atau rule tertentu

5. Dapat dimodifikasi
6. Berbasis pengetahuan dan mekanisme inferensi yang terpisah
7. Output atau luarannya bersifat himbauan
8. Sistem pakar dapat mengaktifkan aturan atau rule secara searah yang sesuai, dituntun oleh dialog dengan pengguna.

### **E. Ruang Lingkup Problem dalam Sistem Pakar**

Aplikasi sistem pakar meliputi beberapa ruang lingkup antara lain :

1. Interpretasi, menghasilkan deskripsi situasi berdasarkan data-data masukan
2. Prediksi, memperkirakan akibat yang mungkin terjadi dari situasi yang ada
3. Diagnosis, menyimpulkan suatu keadaan berdasarkan gejala-gejala yang diberikan
4. Melakukan perancangan (*design*) berdasarkan kendala-kendala yang diberikan
5. Merencanakan (*planning*) tindakan yang dilakukan
6. Memonitor (*monitoring*) hasil pengamatan
7. Menentukan pemecahan (*debugging*) dari suatu kesalahan sistem
8. Melaksanakan rencana perbaikan
9. Melaksanakan instruksi dalam diagnosa, debugging dan perbaikan kinerja
10. Melaksanakan pengendalian pada hasil interpretasi

## F. Perbandingan Sistem Konvensional dan Sistem Pakar

Perbedaan yang dapat ditinjau dari sistem konvensional dibandingkan sistem pakar dapat dilihat pada tabel berikut :

Sistem Konvensional	Sistem Pakar
1. Informasi dan pemrosesannya digabungkan dalam satu program	1. Basis pengetahuan dipisahkan secara jelas dengan mekanisme inferensi
2. Program tidak membuat kesalahan (yang membuat kesalahan yaitu pemrogram)	2. Program dapat berbuat kesalahan
3. Biasanya tidak menjelaskan mengapa data masukan diperlukan atau bagaimana output dihasilkan	3. Penjelasan merupakan bagian terpenting dari semua sistem pakar.
4. Perubahan program sangat menyulitkan	4. Perubahan dalam aturan-aturan mudah untuk dilakukan
5. Sistem hanya bisa beroperasi setelah lengkap atau selesai	5. Sistem dapat beroperasi hanya dengan aturan-aturan yang sedikit (sebagai prototipe awal)
6. Eksekusi dilakukan langkah demi langkah	6. Eksekusi dilakukan dengan menggunakan heuristik dan logika pada seluruh bisnis pengetahuan.
7. Perlu informasi lengkap agar bisa beroperasi	7. Dapat beroperasi dengan informasi yang tidak lengkap atau mengandung ketidakpastian
8. Manipulasi efektif dari basis data yang besar	8. Manipulasi efektif dari basis pengetahuan yang besar.
9. Menggunakan data	9. Menggunakan pengetahuan
10. Tujuan utama : efisiensi	10. Tujuan utama : efektivitas
11. Mudah berurusan dengan data kuantitatif	11. Mudah berurusan dengan data kualitatif
12. Menangkap, menambah dan mendistribusikan akses ke data numerik atau informasi	12. Menangkap, menambah dan mendistribusikan akses ke pertimbangan dan pengetahuan

## G. Membangun Sistem Pakar

Pada dasarnya, ada 4 langkah untuk membangun sebuah sistem pakar antara lain :

1. Analisis

Tujuan analisis adalah mengidentifikasi aplikasi yang potensial. Aplikasi yang mungkin termasuk diagnostik, controller dan lain-lain. Selama analisis, pengembang juga harus menilai kesesuaian pengetahuan rekayasa teknologi untuk aplikasi ini. Anda harus yakin dan bertanya apakah sistem ini dapat bekerja lebih baik. Hal ini berlaku untuk menerapkan semua jenis kecerdasan buatan untuk memecahkan problem.

2. Spesifikasi

Langkah spesifikasi adalah tempat pengembang mengidentifikasi kemampuan sistem pakar. Para pengembang juga harus bekerja dengan para pakar untuk belajar dan merencanakan pengembangan sistem. Ahli adalah manusia yang diidentifikasi sebagai ahli di domain bidang tertentu, seperti psikolog, dokter dan ilmuwan. Pengembang harus membiasakan diri dengan problem sehingga pengembangan sistem dapat dilakukan. Pengembang akan menghabiskan sistem dapat dilakukan. Pengembang akan menghabiskan sejumlah besar waktu dalam fase memperoleh pengetahuan.

Mendefinisikan kemampuan sistem pakar adalah sesuatu yang sangat menantang. Para ahli dapat mengatasi problem dengan cara yang berbeda. Beberapa ahli mungkin membayangkan fungsionalitas yang berbeda pada sebuah sistem daripada seorang pengembang yang lebih memahami keterbatasan dari software. Hal ini juga penting untuk meyakinkan para pakar bahwa tujuan dari sistem pakar adalah bukan menggantikan

para ahli, tetapi mengembangkan pengetahuan dan keahlian seluruh organisasi.

### 3. Pengembangan

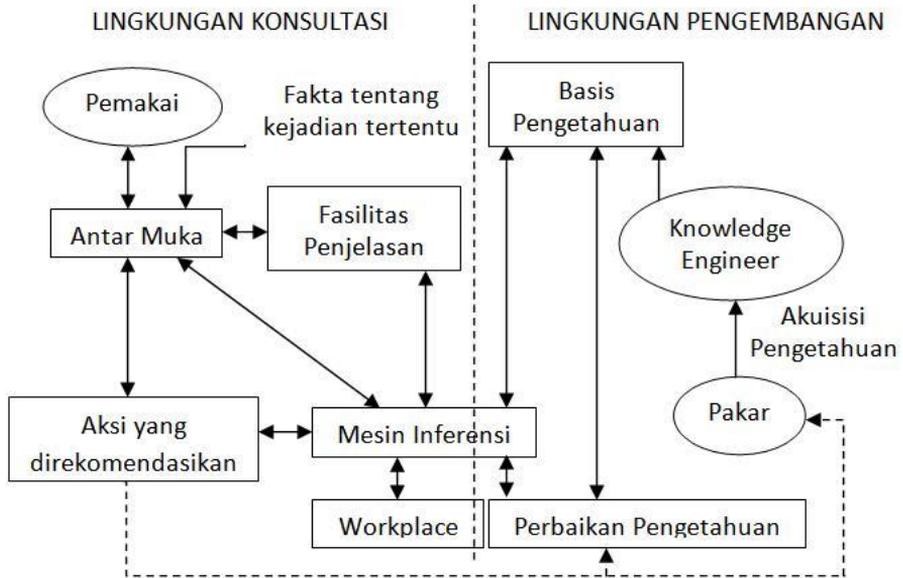
Pada tahap ini, pengembang harus belajar cara ahli melakukan tugas (akuisisi pengetahuan) dalam berbagai kasus. Pada dasarnya, ada tiga jenis kasus dimana pengembang harus berdiskusi dengan ahli yaitu “saat ini”, “historis”, dan “hipotesis”. Kasus saat ini dapat diperoleh dengan mengamati seorang ahli saat melakukan tugas.

### 4. Penyebaran

Setelah program selesai dibuat dan diuji, maka dapat didistribusikan ke masyarakat untuk dapat digunakan secara lebih luas.

## **H. Struktur Sistem Pakar**

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



Keterangan :

1. Masukan pengetahuan, subsistem ini digunakan untuk memasukkan pengetahuan agar bisa dimekanisme oleh komputer dan meletakkannya ke dalam basis pengetahuan dengan format tertentu dalam bentuk representasi pengetahuan. Sumber-sumber pengetahuan bisa diperoleh dari pakar, buku, dokumen, multimedia, basis data, laporan riset khusus dan informasi yang terdapat di web.

2. Basis pengetahuan (*knowledge base*)

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

- a. Fakta, misalnya situasi, kondisi atau perproblemahan yang ada
- b. Rule (antrean), untuk mengarahkan penggunaan pengetahuan dalam memecahkan problem.

3. Mesin inferensi (*inference engine*)

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

4. Daerah Kerja (*Blackboard*)

Untuk merekam hasil sementara yang akan dijadikan sebagai keputusan dan untuk menjelaskan sebuah problem yang sedang terjadi, sistem pakar memerlukan *Blackboard*, yaitu area pada penyimpanan yang berfungsi sebagai basis data. Tiga tipe keputusan yang dapat direkam pada blackboard yaitu :

- a. Rencana, bagaimana menghadapi problem
- b. Agenda : aksi-aksi potensial yang sedang menunggu untuk dieksekusi

c. Penyelesaian : calon aksi yang akan dibangkitkan

5. Antarmuka Pengguna (*User Interface*)

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

6. Subsystem Penjelasan (*Explanation Subsystem/Justifier*)

Berfungsi memberi penjelasan kepada pengguna, bagaimana suatu kesimpulan dapat diambil. Kemampuan seperti ini sangat penting bagi pengguna untuk mengetahui mekanisme pemindahan keahlian pakar maupun dalam pemecahan problem.

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

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

## 8. Pengguna (*User*)

Pada umumnya pengguna sistem pakar bukanlah seorang pakar (*non expert*) yang memerlukan penyelesaian, saran atau pelatihan (*training*) dari berbagai perproblemaman yang ada.

### I. Perambatan Maju (*Forward Chaining*)

Pada penalaran maju, aturan-aturan diuji satu demi satu dalam urutan tertentu, urutan tersebut mungkin berupa aturan ke dalam perangkat aturan atau dapat juga urutan lain yang ditentukan oleh pemakai. Dalam pengujian tersebut sistem pakar berusaha mengevaluasi apakah kondisinya benar atau salah. Apabila kondisi benar, aturan tersebut ditembakkan dan aturan berikutnya diuji. Apabila kondisinya salah, aturan tersebut tidak ditembakkan dan aturan berikutnya diuji. Apabila kondisinya salah, aturan tersebut tidak ditembakkan dan aturan berikutnya diuji.

Suatu aturan mungkin tidak dievaluasi sebagai benar atau salah. Mungkin kondisinya mencakup satu atau beberapa variabel dengan nilai yang tidak diketahui. Dalam hal itu, kondisi aturannya tidak diketahui. Apabila kondisi aturan tidak diketahui, aturan tidak ditembakkan dan aturan berikutnya diuji.

Mekanisme pengujian aturan satu demi satu berlanjut sampai putaran lengkap melalui seluruh perangkat aturan. Biasanya diperlukan lebih dari satu putaran untuk memberikan suatu nilai pada variabel sasaran. Mungkin informasi yang diperlukan untuk mengevaluasi satu aturan dihasilkan oleh aturan lain yang diuji kemudian. Ketika tidak ada lagi aturan yang dapat ditembakkan, maka mekanisme penalaran berhenti. Gambar 2.3 adalah contoh algoritma *forward chaining*

## Forward chaining algorithm

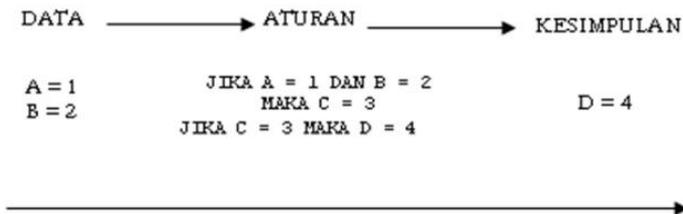
```

function FOL-FC-ASK( $KB, \alpha$ ) returns a substitution or false
  repeat until new is empty
     $new \leftarrow \{ \}$ 
    for each sentence  $r$  in  $KB$  do
       $(p_1 \wedge \dots \wedge p_n \Rightarrow q) \leftarrow \text{STANDARDIZE-APART}(r)$ 
      for each  $\theta$  such that  $(p_1 \wedge \dots \wedge p_n)\theta = (p'_1 \wedge \dots \wedge p'_n)\theta$ 
        for some  $p'_1, \dots, p'_n$  in  $KB$ 
           $q' \leftarrow \text{SUBST}(\theta, q)$ 
          if  $q'$  is not a renaming of a sentence already in  $KB$  or new then do
            add  $q'$  to new
             $\phi \leftarrow \text{UNIFY}(q', \alpha)$ 
            if  $\phi$  is not fail then return  $\phi$ 
    add new to  $KB$ 
  return false

```

Gambar 2.3 algoritma forward chaining

Pada perambatan maju (*forward chaining*) menggunakan himpunan aturan kondisi – aksi. Dalam prosedur ini, data digunakan untuk menentukan aturan mana yang akan dijalankan, kemudian aturan tersebut dijalankan. Prosedur ini cocok digunakan untuk menangani problem pengendalian (*controlling*) dan peramalan (*pragnosis*), contoh :

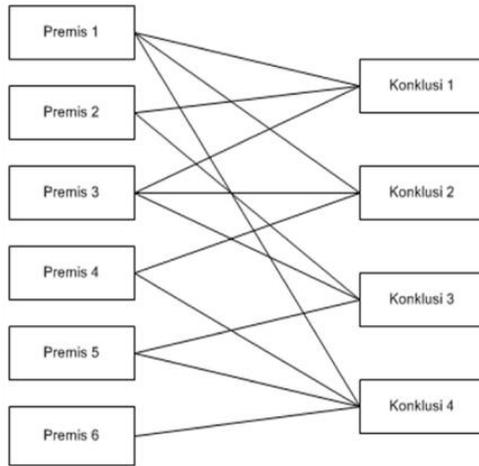


Urutan langkah perambatan maju :

- Tampilkan semua daftar premis
- User memilih premis yang dialami
- Sistem mencari aturan yang premisnya terdiri dari premis-premis yang dipilih oleh user
- Sistem akan menampilkan konklusi

Langkah-langkah algoritma perambatan maju :

- Tampilkan semua daftar premis  
 Premis :  
 Premis 1                      Premis 4  
 Premis 2                      Premis 5  
 Premis 3                      Premis 6
- User memilih premis yang dialami  
 Pilihan user misalnya :  
 Premis 1  
 Premis 2  
 Premis 3  
 Konklusinya yaitu konklusi 1
- Sistem mencari aturan yang premisnya terdiri dari premis-premis yang dipilih oleh user
- Sistem akan menampilkan konklusi dari aturan tersebut.  
 Pilihan user : Premis 1 dan Premis 6  
 Maka konklusinya yaitu : Konklusi 4



Untuk mengetahui apakah suatu fakta yang dialami oleh pengguna itu termasuk konklusi 1, konklusi 2, konklusi 3 atau konklusi 4 atau bahkan bukan salah satu dari konklusi tersebut. User diminta memasukkan premis-premis yang dialami. Sistem dapat memunculkan daftar premis yang mungkin sehingga user dapat memberikan feedback premis mana yang dialami dengan memilih satu atau beberapa dari daftar premis yang tersedia.

Berdasarkan premis-premis yang dipilih maka sistem akan mencari aturan yang sesuai, sehingga akan diperoleh konklusinya.

Contoh Kasus forward chaining :

1. Misalkan diketahui sistem pakar menggunakan 5 buah rule sebagai berikut :

R1 : If (Y and D) then Z

R2 : If (X and B and E) then Y

R3 : If A then X

R4 : If C then L

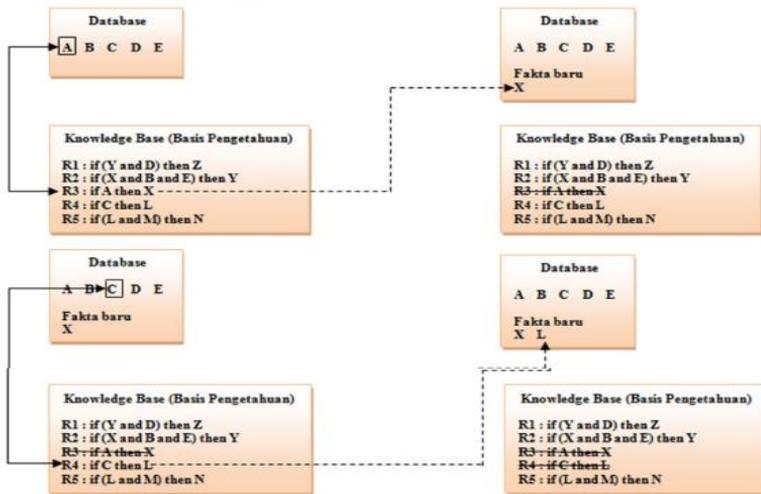
R5 : If (L then M) then N

Fakta-fakta : A, B, C, D dan E bernilai benar

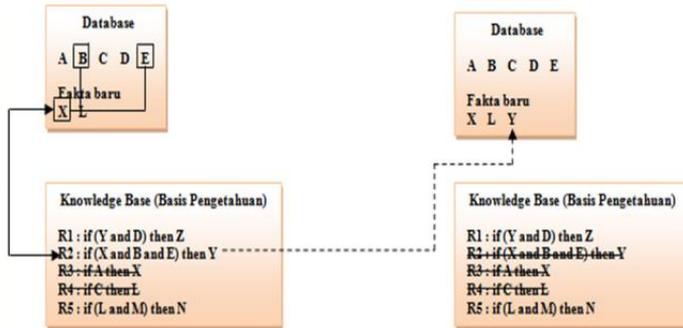
Tujuan : menentukan apakah Z bernilai benar

Langkah-langkahnya sebagai berikut :

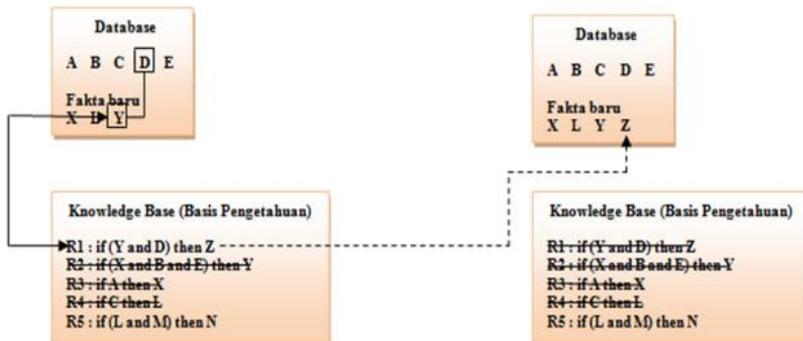
## Iterasi I



## Iterasi 2



## Iterasi ke 3



Sampai disini mekanisme dihentikan karena sudah tidak ada lagi rule yang bisa dieksekusi penelusuran adalah Z bernilai benar (lihat database di bagian fakta baru).

2. Berikut ini adalah beberapa orang yang akan mengambil keputusan untuk investasi :

A : memiliki 100 juta

B : < 30 tahun

C : Pendidikan Sarjana

D : Pendidikan tahunan  $\leq$  400 juta

E : Investasi di saham

F : Investasi di saham pertumbuhan

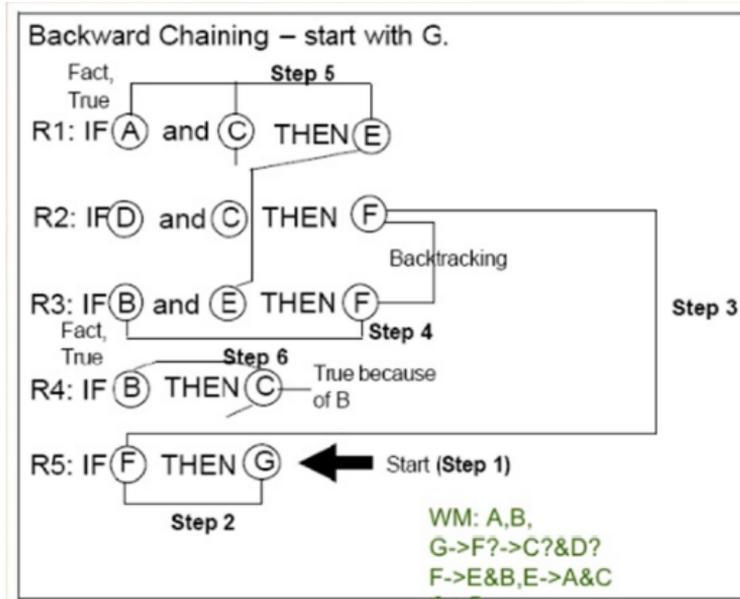
G : Investasi di saham IBM

Fakta yang diketahui :

Seseorang memiliki 100 juta dan berumur 25 tahun, ingin meminta saran apakah sebaiknya dia berinvestasi di saham IBM atau tidak.

Rule yang diketahui adalah :

1. Apabila memiliki 100 juta dan memiliki pendidikan sarjana maka cocok investasi di saham
2. Apabila pendapatan tahunan  $\leq$ 400 juta dan pendidikannya sarjana maka cocok investasi di saham pertumbuhan
3. Apabila umur < 30 tahun dan sudah investasi di saham maka cocok investasi di saham pertumbuhan
4. Apabila umur < 30 tahun maka pasti pendidikannya sarjana
5. Apabila investasi di saham pendidikan maka pasti juga investasi di saham IBM



### J. Perambatan Mundur (*Backward Chaining*)

Pada penalaran mundur (*Backward Chaining*), inference engine memilih suatu aturan dan menganggapnya sebagai problem yang harus diselesaikan. Dengan menggunakan perangkat aturan inference engine mulai mengevaluasi dari variabel sasaran, kemudian diikuti dengan pemilihan salah satu subproblem untuk dievaluasi, dan subproblem yang terpilih akan dievaluasi sebagai problem baru. Inference engine terus mencari subproblem untuk menjadi problem baru yang akan dievaluasi sampai dengan tidak ada lagi subproblem yang ditemui. Penalaran maju bergerak lebih cepat dari penalaran mundur karena penalaran mundur tidak harus mempertimbangkan semua aturan dan tidak membuat beberapa putaran melalui perangkat aturan. Penalaran mundur sangat sesuai digunakan apabila :

- a. Terdapat variabel sasaran berganda (*multiple sasaran variable*)
- b. Terdapat banyak aturan.
- c. Semua atau hampir semua aturan tidak harus diuji dalam mekanisme mencapai pemecahan

Gambar 2.4 adalah contoh algoritma backward chaining

```

function FOL-BC-ASK(KB, goals,  $\theta$ ) returns a set of substitutions
  inputs: KB, a knowledge base
           goals, a list of conjuncts forming a query
            $\theta$ , the current substitution, initially the empty substitution { }
  local variables: ans, a set of substitutions, initially empty
  if goals is empty then return { $\theta$ }
   $q' \leftarrow \text{SUBST}(\theta, \text{FIRST}(\textit{goals}))$ 
  for each r in KB where  $\text{STANDARDIZE-APART}(r) = (p_1 \wedge \dots \wedge p_n \Rightarrow q)$ 
    and  $\theta' \leftarrow \text{UNIFY}(q, q')$  succeeds
     $\textit{ans} \leftarrow \text{FOL-BC-ASK}(\textit{KB}, [p_1, \dots, p_n | \text{REST}(\textit{goals})], \text{COMPOSE}(\theta', \theta)) \cup \textit{ans}$ 
  return ans

```

Sumber:turban:2005

Gambar 2.4 Algoritma backward chaining

## K. Contoh Kasus Sistem Pakar

Penelitian terkait sistem pakar dalam menentukan kerusakan notebook dengan menggunakan forward chaining. Beberapa contoh kerusakan pada yang biasa terjadi pada notebook, ditunjukkan pada tabel berikut :

Tabel 2.2 Kerusakan pada notebook

No	Gejala	Kerusakaan hardware
1	Layar tidak tampil gambar	Problem pada LCD atau kabel fleksibel
2	Blue screen saat loading	Ram, hardisk
3	Menyala tapi keluar garis	Problem pada LCD atau kabel fleksibel
4	Gambar tidak simetris	Problem pada LCD atau kabel fleksibel
5	Layar hanya bewarna putih	Problem pada LCD atau kabel fleksibel
6	Restart sendiri	Problem pada LCD atau kabel fleksibel
7	Baterai tidak mau di charge	Adaptor
8	Layar tidak tampil gambar	Problem pada LCD atau kabel fleksibel
9	Blue screen saat loading	Ada problem pada Harddisk atau ram
10	Menyala tapi keluar garis	Problem pada LCD atau kabel fleksibel
11	Gambar tidak simetris	Problem pada LCD atau kabel fleksibel
12	Kursor tidak stabil	Sensor kursor sudah tidak sensitif
13	Beberapa tuts tidak berfungsi	Keyboard rusak
14	Loading data lambat	Problem pada harddisk
15	Laptop posisi di charge kemudian mati	Baterai atau ada problem pada pendingin
16	Tidak bisa membaca dvd	Optical drive kotor atau rusak
17	Tidak bisa masuk windows	Harddisk error
18	Harddisk berbunyi keras	Kemungkina terjadi bad sector
19	Terdapat suara nyaring saat dihidupkan	Terdapat problem pada ram
20	Usb tidak terdeteksi	Slot usb kotor atau ada problem dengan harddisk
21	Notebook cepat panas	Kipas pendingin atau heatsink tidak bekerja dengan normal

## 1. Tahap Analisis

Akuisisi pengetahuan dalam penelitian ini berasal dari studi pustaka yang merupakan teknik penelusuran dengan melakukan penelusuran data lewat literatur-literatur yang terkait dengan notebook misalnya buku-buku referensi, artikel, materi diklat, internet. Akuisisi pengetahuan diorganisasi dan distrukturisasi menjadi aturan-aturan detail dan jelas agar komputer dapat mengakses data yang diperlukan untuk pengambilan kesimpulan.

Basis pengetahuan merupakan sekumpulan pengetahuan yang dihubungkan dengan sejumlah perproblemaman yang digunakan dalam sistem kecerdasan buatan. Basis pengetahuan ini merupakan analisa data yang akan digunakan dalam pembangunan sistem. Basis sistem pakar Deteksi kerusakan notebook berbasis web ditunjukkan seperti pada tabel berikut

Tabel Basis Pengetahuan

No.	Kerusakan	Deskripsi	Penyelesaian
1.	Kerusakan Pada DVD Room	Tidak mau membaca DVD Room, indikator DVD Room off.	Dalam problem ini perlu adanya pengecekan berkali-kali terhadap DVD Room , apabila tidak juga terbaca oleh notebook maka harus diadakan pembongkaran untuk mengganti DVD Roomnya,atau bawa ke service center resminya
2.	Kerusakan Pada Wifi	Jaringan tidak terdeteksi,  Koneksi sering putus	Pengecekan pada software dan hardware apabila terdapat kerusakan
3.	Kerusakan Pada Lan Card	Transfer data lambat,file transfer sering rusak	Pengecekan pada port land an kabel lan
4.	Kerusakan Pada Bluetooth	Perangkat Bluetooth tidak terdeteksi dan file data transfer rusak	Pengecekan pada hardware dan perangkat lunak
5.	Kerusakan Pada Hardisk	Loading data / System lambat, berbunyi tidak normal, tidak bisa masuk operating system, belum sampai login operating system sudah restart sendiri.	Dalam hal ini kerusakan pada harddisk maka tidak dapat masuk ke operating system, harus dilakukan pembongkaran dan format data di harddisk dan coba dipasang kembali apa bila masih dapat digunakan, apabila tidak harus dilakukan penggantian hardisk.

6.	Kerusakan Pada LCD	Layar tidak tampil gambar, menyala tapi keluar garis-garis vertikal, tampak blok hitam, dan gambar tidak simetris / acak.	Matikan laptop dengan hard turn off (tekan tombol power sampai mati), lalu lepas baterai dan kabel chargernya, terus tekan tombol power off beberapa detik untuk menghilangkan muatan kapasitansi, terus pasang baterai, nyalakan notebook. Apabila masih muncul, maka kemungkinan ada kerusakan hardware, atau sebaiknya bawa ke service center resmi.
7.	Kerusakan Pada Mainboard	Dihidupkan agak sulit, baterai tidak mau dicharge, Mati Total. Indikator charger menyala, setelah dicarge lampu indikator pada charger mati (konslet). Berarti terjadi arus balik pada powernya. Kerusakan ini sering terjadi.	Kemungkinan besar hal itu terjadi karena beberapa jenis kegagalan motherboard. Anda dapat mencoba reseating / mengganti penyimpanan . Apabila tidak membantu, berarti kerusakan motherboard. Dalam kebanyakan kasus, sangat sulit mengganti motherboard, lebih baik membeli notebook baru.

8.	Kerusakan Pada Baterai	Baterai tidak mau di charge, tidak ada indikator masuk power, laptop di charge posisi hidup malah kemudian mati. Layar bergetar tidak stabil	Kerusakan baterai. Apabila baterai benar-benar tidak terisi sedikitpun, coba gunakan dulu baterai saja. Apabila tidak membantu, keluarkan baterai dan coba jalankan notebook hanya dari AC adaptor. Apabila notebook berjalan dengan baik dari adaptor AC tetapi tidak mengisi baterai sama sekali, kemungkinan besar baterai rusak atau problem motherboard. Cobalah mengganti baterai dulu. Lepas konektor daya DC. Apabila pengisian baterai berhenti setelah Anda mengatur posisi ujung adaptor AC di dalam konektor, berarti DC jack berproblem.
9.	Kerusakan Pada Webcam	Gambar tidak jelas dan perangkat keras tidak terdeteksi dan gambar buram	Cek pada lensa webcam dan pengecekan hardware dan driver webcam cek kualitas video yang dihasilkan
10.	Kerusakan Pada Prosesor	Booting sendiri, hang cepat panas dan notebook terasa lambat	Cek kinerja notebook apakah lambat atau normal dan perhatikan suhu notebook
11.	Kerusakan Pada MMC	Data file transfer rusak, mmc tidak terdeteksi karena slot kotor	Pengecekan pada driver dan hardware apabila terdapat kerusakan
12.	Kerusakan Pada Mouse	Diakibatkan oleh bidang yang kotor dan bagian touchpad panas	Kemungkinan kotor pada bidang pad mouse, Cek hardware dan driver mouse

13.	Kerusakan Pada Keyboard	Beberapa tuts pada keyboard tidak berfungsi, cursor berjalan tidak stabil / bergerak sendiri. Kerusakan yang lebih parah biasanya terjadi konslet dan ini menyebabkan notebook restart-restart terus.	Kemungkinan besar beberapa tombol keyboard pada keyboard gagal. keyboard tidak dapat diperbaiki. Apabila beberapa tombol berhenti bekerja, Anda harus mengganti keyboard.
-----	-------------------------	---	---

Model pengaplikasian kepakaran untuk sistem pakar Deteksi kerusakan notebook web base ini menggunakan bentuk aturan penerapan yang diaplikasikan dalam bentuk jika-maka (if-then). Tabel kerusakan mendeskripsikan tentang Deteksi kerusakan notebook web base. Tabel kerusakan ditunjukkan seperti pada tabel berikut :

Tabel Kerusakan

<b>No.</b>	<b>Kode Kerusakan</b>	<b>Nama Kerusakan</b>
1.	K001	Kerusakan pada DVD RW
2.	K002	Kerusakan pada Wifi
3.	K003	Kerusakan pada Land Card
4.	K004	Kerusakan pada Bluetooth
5.	K005	Kerusakan pada Hardisk
6.	K006	Kerusakan pada LCD
7.	K007	Kerusakan pada Mainboard
8.	K008	Kerusakan pada Baterai
9.	K009	Kerusakan pada Webcam
10.	K010	Kerusakan pada Prosesor
11.	K011	Kerusakan pada MMC
12.	K012	Kerusakan pada Mouse
13.	K013	Kerusakan pada Keyboard

Tabel penyelesaian menjelaskan tentang penyelesaian Deteksi kerusakan notebook berbasis web. Tabel penyelesaian ditunjukkan pada tabel berikut :

Tabel Penyelesaian

No.	Kode Kerusakan	Deskripsi	Penyelesaian
1.	K001	Tidak mau membaca DVD Room, indikator DVD Room off.	Dalam problem ini perlu adanya pengecekan berkali-kali terhadap DVD Room , apabila tidak juga terbaca oleh notebook maka harus diadakan pembongkaran untuk mengganti DVD Roomnya,atau bawa ke service center resminya
2.	K002	Jaringan tidak terdeteksi,  Koneksi sering putus	Pengecekan pada software dan hardware apabila terdapat kerusakan
3.	K003	Transfer data lambat,file transfer sering rusak	Pengecekan pada port land an kabel lan
4.	K004	Perangkat Bluetooth tidak terdeteksi dan file data transfer rusak	Pengecekan pada hardware dan perangkat lunak atau driver
5.	K005	Loading data / System lambat, berbunyi tidak normal, tidak bisa masuk operating system, belum sampai login operating system sudah restart sendiri.	Dalam hal ini kerusakan pada harddisk maka tidak dapat masuk ke operating system, harus dilakukan pembongkaran dan format data di harddisk dan coba dipasang kembali apa bila masih dapat digunakan, apabila tidak harus dilakukan penggantian hardisk.
6.	K006	Layar tidak tampil gambar, menyala tapi keluar garis-garis vertikal, tampak blok hitam, dan gambar tidak simetris / acak.	Matikan laptop dengan hard turn off (tekan tombol power sampai mati), lalu lepas baterai dan kabel chargernya, terus tekan tombol power off beberapa detik untuk menghilangkan muatan

No.	Kode Kerusakan	Deskripsi	Penyelesaian
			kapasitansi, terus pasang baterai, nyalakan notebook. Apabila masih muncul, maka kemungkinan ada kerusakan hardware, atau sebaiknya bawa ke service center resmi.
7.	K007	Dihidupkan agak sulit, baterai tidak mau dicharge, Mati Total. Indikator charger menyala, setelah dicarge lampu indikator pada charger mati (konslet). Berarti terjadi arus balik pada powernya. Kerusakan ini sering terjadi.	Kemungkinan besar hal itu terjadi karena beberapa jenis kegagalan motherboard. Anda dapat mencoba reseating / mengganti penyimpanan . Apabila tidak membantu, berarti kerusakan motherboard. Dalam kebanyakan kasus, sangat sulit mengganti motherboard, lebih baik membeli notebook baru.
8.	K008	Baterai tidak mau di charge, tidak ada indikator masuk power, laptop di charge posisi hidup malah kemudian mati. Layar bergetar tidak stabil	Kerusakan baterai. Apabila baterai benar-benar tidak terisi sedikitpun, coba gunakan dulu baterai saja. Apabila tidak membantu, keluarkan baterai dan coba jalankan notebook hanya dari AC adaptor. Apabila notebook berjalan dengan baik dari adaptor AC tetapi tidak mengisi baterai sama sekali, kemungkinan besar baterai rusak atau problem motherboard. Cobalah mengganti baterai dulu. Lepas konektor daya DC. Apabila pengisian baterai berhenti setelah Anda mengatur posisi ujung

No.	Kode Kerusakan	Deskripsi	Penyelesaian
			adaptor AC di dalam konektor, berarti DC jack berproblem.
9.	K009	Gambar tidak jelas dan perangkat keras tidak terdeteksi dan gambar buram	pengecekan hardware webcam dan cek driver
10.	K010	Booting sendiri, hang cepat panas dan notebook terasa lambat	Cek kinerja notebook apakah lambat atau normal dan perhatikan suhu notebook
11.	K011	Data file transfer rusak, mmc tidak terdeteksi dan loading drive mmc lambat	Pengecekan pada driver dan hardware apabila terdapat kerusakan
12.	K012	Kursor hilang bagian trackpad panas dan pointer bergerak tidak beraturan	Cek hardware dan driver mouse
13.	K013	Beberapa tuts pada keyboard tidak berfungsi, cursor berjalan tidak stabil / bergerak sendiri. Kerusakan yang lebih parah biasanya terjadi konslet dan ini menyebabkan notebook restart-restart terus.	Kemungkinan besar beberapa tombol keyboard pada keyboard gagal. keyboard tidak dapat diperbaiki. Apabila beberapa tombol berhenti bekerja, Anda harus mengganti keyboard.

Kaidah penerapan biasanya dicatat kedalam format *IF-THEN*, kaidah ini dapat disebutkan sebagai hubungan keterlibatan dua bagian yaitu bagian premis (apabila) dan bagian simpulan (maka), apabila bagian premis dipenuhi, maka bagian simpulan juga akan bernilai benar.

Tabel keputusan sistem pakar Deteksi kerusakan notebook web base ditunjukkan pada table berikut:

Tabel Keputusan

Kode	Gejala Kerusakan	Kode Kerusakan
G001	Tidak bisa baca DVD	K001
G002	Keluar suara pembacaan yang berat	K001
G003	LED DVD tidak nyala hijau	K001
G004	Tidak mendapatkan sinyal wifi	K002
G005	Hidup tapi tidak bisa konek	K002
G006	LED wifi tidak nyala	K002
G007	Hardware tidak terdeteksi	K003
G008	Tidak ada keterangan LAN hidup	K003
G009	LED LAN tidak hidup	K003
G010	Bluetooth tidak terdeteksi	K004
G011	Tidak bisa mengirim data lewat bluetooth	K004
G012	Data yang dikirim rusak	K004
G013	Bluescreen saat loading	K005
G014	Hardisk tidak terdeteksi	K005
G015	Kinerja lambat	K005
G016	Layar tidak memunculkan image	K006
G017	Nyala tetapi muncul garis vertical	K006
G018	Terlihat block warna hitam	K006
G019	Notebook tidak bisa dinyalakan	K007
G020	LED supply daya tidak nyala	K007
G021	Bunyi tit panjang	K007
G022	Baterai tidak mau dicarge	K008
G023	LED indikator tidak menyala	K008
G024	AC adaptor panas	K008
G025	LED webcam tidak hidup	K009
G026	Webcam aktif tapi tidak bisa merekam	K009
G027	Gambar tidak jelas atau buram	K009
G028	Notebook mati sendiri	K010
G029	Notebook berhenti bekerja / hang	K010
G030	Notebook restart sendiri	K010
G031	Kinerja lambat	K011
G032	Data MMC tidak terdeteksi	K011
G033	Data putus saat melakukan copy paste	K011

Kode	Gejala Kerusakan	Kode Kerusakan
G034	Klik kanan kiri tidak bias	K012
G035	Pointer tidak keluar	K012
G036	Mouse pad tidak respon	K012
G037	Tuts keyboard tidak berfungsi	K013
G038	Huruf yang ditekan keluar berbeda	K013
G039	Beep 1 panjang dan 3 kali pendek	K013

Pada tiap-tiap area gejala, terdapat juga 8 bentuk aturan penerapan gejala kerusakan dalam format *IF-THEN rules*. Table 3.5 berikut ini merupakan tabel pembentukan aturan dari sistem pakar yang akan dikembangkan.

Table Pembentukan *Rule*

<b>RULE</b>	<b>IF</b>	<b>THEN</b>
R1	G001, G002, G003	K001
R2	G004, G005, G006	K002
R3	G007, G008, G009	K003
R4	G010, G011, G012	K004
R5	G013, G014, G015	K005
R6	G016, G017, G018	K006
R7	G019, G020, G021	K007
R8	G022, G023, G024	K008
R9	G025, G026, G027	K009
R10	G028, G029, G030	K010
R11	G031, G032, G033	K011
R12	G034, G035, G036	K012
R13	G037, G038, G039	K013

### **Perancangan Pohon Keputusan Deteksi Kerusakan Notebook**

Skema pohon keputusan merupakan sesuatu bagan yang dipergunakan untuk menghasilkan sesuatu sistem pakar, di dalam skema pohon keputusan tersebut akan dicari penyelesaian hasil akhir dari setiap pemeriksaan aturan. Skema pohon keputusan akan memudahkan untuk menyusun *knowledge base* dan tata cara serta menentukan faktor ketetapan dari setiap perwujudan deteksi gejala pada kerusakan notebook.

Hasil analisis menggunakan weka untuk pembentukan struktur pohon.

=== Run information ===

Scheme: weka.classifiers.trees.J48 -C 0.25 -M 2

Relation: DataTraining

Instances: 44

Attributes: 64

Loading lambat  
bluescreen  
Led HDD Mati  
Layar Blank  
Not Charge  
Overheat Batrey  
Bunyi Apnormal  
Bunyi Normal  
Harddisk tak terdeteksi  
Led Power tak NYala  
Keluar Asap  
Garis vertikal  
NYala Lalu Blank  
Keluar Asap  
Indikator Not Charge  
DaYa tidak 100  
Not Boot  
Lambat  
Tit Panjang  
Dead Pixel  
Garis vertikal  
Gambar Buram  
LCD tak stabil  
Adaptor Panas  
Batery Low  
DVD tak terbaca  
Sulit Baca DVD  
Video tak terdeteksi  
Video buram  
webcam no recording  
led webcam mati  
gambar LCD hanya separuh

Led DVD tak nyala warna Hijau  
peringatan overheat  
waktu booting lama  
tiba-tiba restart  
hang  
tiba-tiba mati  
tak keluar huruf/angka di layar  
huruf keluar tak sesuai  
beep panjang dan pendek  
respon terlalu cepat  
huruf/angka keluar sendiri  
lampu led mati  
tak tampil sinyal wifi  
tak bisa konek wifi  
wifi hidup mati  
tanda seru koneksi wifi  
koneksi tak stabil  
kursor hilang  
kursor kurang respon  
kursor berpindah sendiri  
klik mouse tak normal  
pergerakan mouse berat  
area touchpad panas  
hardware lan tak terdeteksi  
sambungan jaringan sering putus  
tak bisa tersambung ke jaringan  
MMC tak terbaca  
terdeteksi tetapi tak tampil sebagai media simpan  
MMC tidak terdeteksi  
transfer data lambat  
file data transfer rusak  
Kerusakan

Test mode: evaluate on training data

=== Classifier model (full training set) ===

J48 pruned tree

-----

- Led DVD tak nyala warna Hijau = Tidak
  - | tak bisa konek wifi = Tidak
    - | | hardware lan tak terdeteksi = Tidak
      - | | | file data transfer rusak = Tidak
        - | | | | Not Boot = Ya: Hardisk (3.0)
        - | | | | Not Boot = Tidak
        - | | | | | bluescreen = Tidak
        - | | | | | Layar Blank = Tidak
        - | | | | | Led HDD Mati = Tidak
        - | | | | | Not Charge = Tidak
        - | | | | | Video buram = Tidak
        - | | | | | | led webcam mati = Tidak
        - | | | | | | tiba-tiba mati = Tidak
        - | | | | | | | Lambat = Tidak
        - | | | | | | | | kursor hilang = Tidak: keyboard (8.0/5.0)
        - | | | | | | | | kursor hilang = Ya: mouse (2.0)
        - | | | | | | | | Lambat = Ya: mmc (3.0/1.0)
        - | | | | | | | | tiba-tiba mati = Ya: Processor (2.0)
        - | | | | | | | | led webcam mati = Ya: Webcam (2.0/1.0)
        - | | | | | | | | Video buram = Ya: Webcam (2.0)
        - | | | | | | | | Not Charge = Ya: Bateray (2.0)
        - | | | | | | | | Led HDD Mati = Ya: Mainboard (2.0)
        - | | | | | | | | Layar Blank = Ya: LCD (3.0/1.0)
        - | | | | | | | | bluescreen = Ya: Hardisk (3.0/1.0)
        - | | | | | | | | file data transfer rusak = Ya: bloetooth (3.0)
        - | | | | | | | | hardware lan tak terdeteksi = Ya: lan card (3.0)
        - | | | | | | | | tak bisa konek wifi = Ya: wifi (3.0)
- Led DVD tak nyala warna Hijau = Ya: DVD RW (3.0)

Number of Leaves : 15

Size of the tree : 29

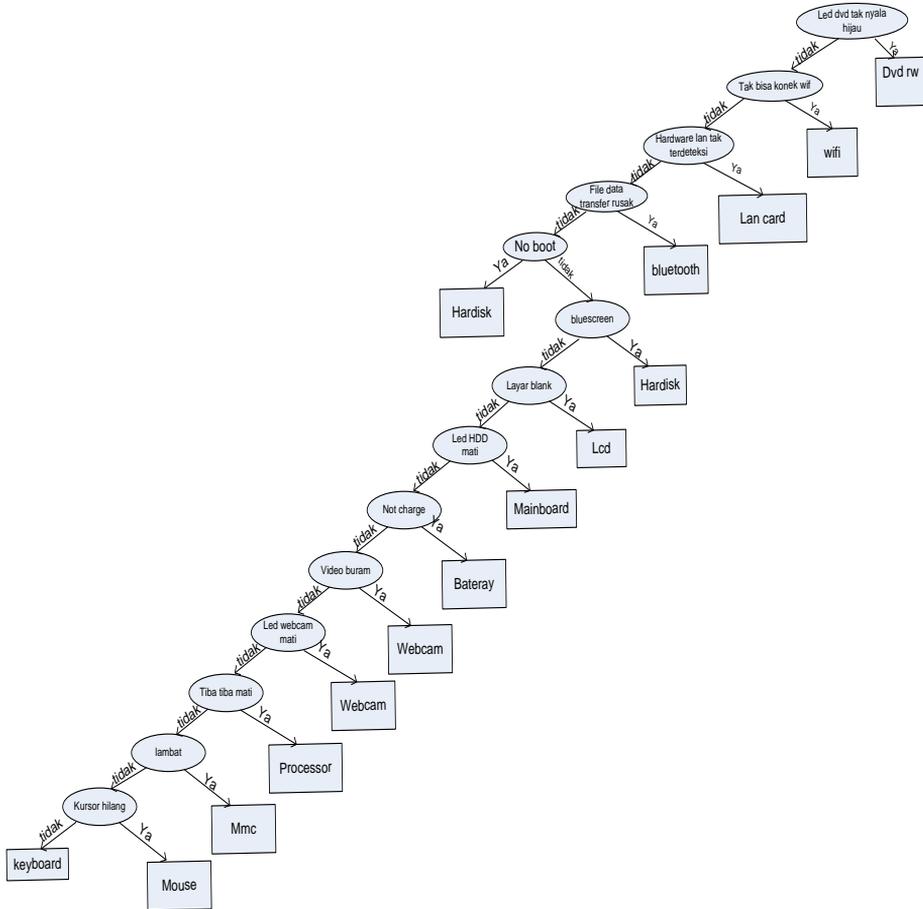
Time taken to build model: 0.03 seconds

==== Evaluation on training set ====

=== Summary ===

Correctly Classified Instances	35	79.5455 %
Incorrectly Classified Instances	9	20.4545 %
Kappa statistic		0.779
Mean absolute error		0.0357
Root mean squared error		0.1336
Relative absolute error		26.9653 %
Root relative squared error		51.938 %
Total Number of Instances	44	

Gambar pohon keputusan hasil dari mekanisme weka dengan menggunakan rule j48



Gambar Pohon Keputusan Deteksi Kerusakan Notebook

## 6. Rule/Aturan Hasil Dari Algoritma Decicion Tree j48.

Rule yang dihasilkan dari algoritma decicion tree RJ48 adalah sebagai berikut :

1. If “Led DVD Tidak nyala Hijau” = “Ya” Then Kerusakan” = “DVD ROM”
2. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” Then Kerusakan = “Wifi”
3. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD”
4. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth”
5. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk”
6. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk”
7. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD”
8. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard”

9. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not charge” = “ya” and then kerusakan = “Baterai”
10. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not charge” = “ya” and then kerusakan = “Baterai” AND “Video buram” = ”Ya” and then kerusakan = “Webcam”
11. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not charge” = “ya” and then kerusakan = “Baterai” AND “Video buram” = ”Ya” and then kerusakan = “Webcam” AND “LED Webcam mati” = “Ya” and then kerusakan = “Webcam”
12. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not

- charge” = “ya” and then kerusakan = “Baterai” AND “Video buram” = ”Ya” and then kerusakan = “Webcam” AND “LED Webcam mati” = “Ya” and then kerusakan = “Webcam” AND “Tiba-Tiba mati” = “Ya” and then kerusakan = “Processor”
13. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not charge” = “ya” and then kerusakan = “Baterai” AND “Video buram” = ”Ya” and then kerusakan = “Webcam” AND “LED Webcam mati” = “Ya” and then kerusakan = “Webcam” AND “Tiba-Tiba mati” = “Ya” and then kerusakan = “Processor” AND “Lambat” = “Ya” and then kerusakan = ”MMC”
  14. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not charge” = “ya” and then kerusakan = “Baterai” AND “Video buram” = ”Ya” and then kerusakan = “Webcam” AND “LED Webcam mati” = “Ya” and then kerusakan = “Webcam” AND “Tiba-Tiba mati” = “Ya” and then kerusakan = “Processor” AND “Lambat” = “Ya” and then kerusakan = ”MMC” AND “Kursor Hilang” = “YA” AND then kerusakan = “mouse”
  15. If “Led DVD Tidak Nyala Hijau” = “Tidak” AND “Tak Bisa Konek Wifi” = “Ya” AND Then Kerusakan = “Wifi” AND “Hardware Lan Tidak Terdeteksi” = “Ya” Then Kerusakan = “LAN CARD” AND “file data transfer rusak” = “Ya” then kerusakan = “Bluetooth” AND “no boot” = “ya” and then kerusakan = “Harddisk” AND “No Boot” = “Tidak” AND “bluescreen” = “Ya” and then kerusakan = “Harddisk” AND “Layar blank” = “ya” and then kerusakan = “LCD” AND “LED

Hdd mati” = “Ya” and then kerusakan = “ Mainboard” and “Not charge” = “ya” and then kerusakan = “Baterai” AND “Video buram” = ”Ya” and then kerusakan = “Webcam” AND “LED Webcam mati” = “Ya” and then kerusakan = “Webcam” AND “ Tiba-Tiba mati” = “Ya” and then kerusakan = “Processor” AND “Lambat” = “Ya” and then kerusakan = ”MMC” AND “Kursor Hilang” = “YA” AND then kerusakan = “mouse” AND “Kursor Hilang” = “Tidak” and then kerusakan = ”Keyboard”

#### **L. Latihan 4**

1. Jelaskan, apakah yang dimaksud dengan sistem pakar ?
2. Apasajakah kelebihan dan manfaat dari sistem pakar ?
3. Sebutkan 8 ciri-ciri sistem pakar !
4. Apasajakah beberapa ruang lingkup dari Aplikasi sistem pakar ?
5. Jelaskan perbedaan antara sistem konvensional dan sistem pakar!
6. Sistem pakar terdiri dari 2 jenis yaitu forward chaining dan backward chaining, jelaskan maksud dari kedua jenis sistem pakar tersebut !

# BAB 5

## Probabilitas Markov

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

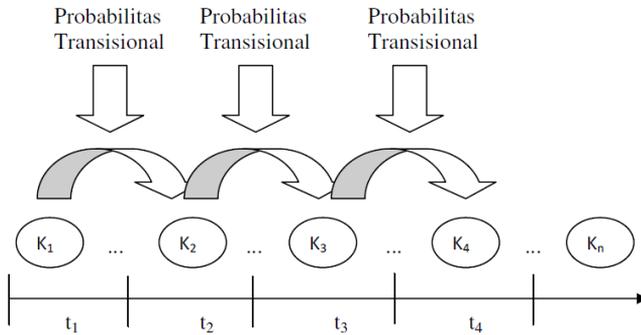
- ❖ Menjelaskan tentang definisi probabilitas Markov
- ❖ Menjelaskan tentang kategori dan analisis golongan Markov
- ❖ Menjelaskan tentang aplikasi probabilitas Markov

### A. Definisi Probabilitas Markov

Rantai markov adalah cara kalkulasi yang digunakan dalam permodelan untuk membantu memprediksikan peralihan yang kemungkinan terjadi di waktu mendatang. Peralihan tersebut diwakili dalam vektor yang berfungsi di masa tertentu, sehingga perlu menyimpan nilai dari vektor keadaan pada masing-masing waktu tertentu.

Rantai markov baru diperkenalkan sekitar tahun 1907 oleh seorang matematikawan yang berasal dari Rusia yaitu Andrey A. Markov (1856-1922) yaitu:

Untuk setiap waktu  $t$ , ketika kejadian adalah  $K_t$  dan seluruh kejadian sebelumnya adalah  $K_{t(j)}, \dots, K_{t(j-n)}$  yang terjadi dari mekanisme yang diketahui, probabilitas seluruh kejadian yang akan datang  $K_{t(j)}$  hanya bergantung pada kejadian  $K_{t(j-1)}$  dan tidak bergantung pada kejadian-kejadian sebelumnya yaitu  $K_{t(j-2)}, K_{t(j-3)} \dots$  dan  $K_{t(j-n)}$ . Gambaran mengenai rantai markov dapat dilihat sebagai berikut :



Gambar Peristiwa dalam Rantai Markov

Gerakan-gerakan dari beberapa variabel di masa yang akan datang bisa diprediksi berdasarkan gerakan-gerakan variabel tersebut di masa lalu.  $K_{t4}$  dipengaruhi oleh kejadian  $K_{t3}$ .  $K_{t3}$  dipengaruhi oleh kejadian  $K_{t2}$  dan demikian seterusnya dimana perubahan ini terjadi karena peranan probabilitas transisi (*transition probability*).

Kejadian  $K_{t2}$  misalnya, tidak akan mempengaruhi kejadian  $K_{t4}$ .

Kejadian-kejadian diatas sifatnya berantai. Oleh karena itu, teori ini dikenal dengan nama Rantai Markov. Dengan demikian, Rantai Markov akan menjelaskan gerakan-gerakan variabel tersebut di masa kini. Secara matematis dapat ditulis :

$$K_{t(j)} = P \times K_{t(j-1)}$$

Dimana

$K_{t(j)}$  = peluang kejadian pada  $t(j)$

$P$  = probabilitas transisional

$t(j)$  = waktu ke- $j$

Peluang kejadian  $K_{t(j)}$  dinyatakan dalam bentuk vektor sehingga seluruh selnya akan selalu 100%.

## **B. Kategori Golongan Markov**

Rantai markov secara umum dapat digolongkan menjadi dua yaitu :

1. Discrete markov chain

Rantai markov dikatakan diskrit apabila perpindahan keadaan terjadi dengan interval waktu distrit yang tetap.

2. Continous markov chain

Rantai markov dikatakan kontinue apabila perpindahan keadaan terjadi dengan rentang waktu dengan variabel random yang continue.

## **C. Analisis Golongan Markov**

Dalam kenyataannya, pengaplikasian analisis Markov dapat dibilang cukup terbatas karena sulit menemukan perproblem yang memenuhi semua sifat yang diperlukan untuk analisis Markov, terutama persyaratan probabilitas transisi harus konstan sepanjang waktu. Mekanisme Rantai Markov terdiri dari dua prosedur, yaitu menyusun matriks probabilitas transisi kemudian menghitung market share di waktu yang akan datang.

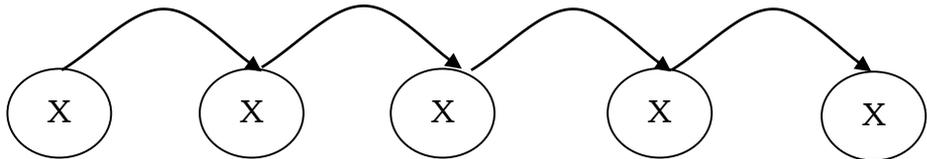
Contoh probabilitas transisi adalah pergantian yang mungkin dilakukan karena faktor periklanan, promosi khusus, harga, ketidakpuasan, dan lain-lain. Hal ini bermanfaat karena jumlah market share-nya di periode yang akan datang dapat diperhitungkan sejak dini.

Untuk dapat menerapkan analisis Markov ke dalam suatu kasus, ada beberapa syarat yang harus dipenuhi :

1. Jumlah probabilitas transisi untuk suatu keadaan awal dari sistem sama dengan 1.
2. Probabilitas tersebut berlaku untuk semua partisipan dalam sistem

3. Probabilitas transisi konstan sepanjang waktu
4. Kondisi merupakan kondisi sepanjang waktu

Prosedur ini banyak digunakan untuk pengambilan keputusan, namun sebenarnya hanya memberikan informasi bagi pengambil keputusan untuk memperbaiki keputusannya, bukan untuk memberi penyelesaian.



Langkah-langkah untuk menyelesaikan perhitungan probabilitas pada periode waktu tertentu menggunakan rantai Markov adalah :

1. Buat matriks transisi dari probabilitas yang diketahui
2. Lakukan operasi perkalian matriks dari probabilitas waktu sebelumnya dengan matriks transisi :  
Matriks periode ke- $n$  = matriks periode ke- $n + 2$  \* matriks transisi
3. Ulangi mekanisme yang sama sampai menemukan probabilitas yang hendak dicari

Dalam perhitungan probabilitas yang terus berulang, akan ditemui pada periode tertentu, bahwa probabilitas hasil perhitungan kedua matriks itu bernilai sama/tetap/tidak berubah apabila dihitung untuk periode-periode selanjutnya. Kondisi ini disebut dengan probabilitas keadaan tetap (*steady state probability*). Langkah yang dikerjakan untuk menemukan probabilitas keadaan tetap adalah sama seperti langkah untuk menghitung probabilitas pada kondisi tertentu.

Nilai probabilitas pada periode-periode selanjutnya setelah bertemu dengan probabilitas keadaan tetap ini akan sama. Karenanya, probabilitas ini dapat digunakan sebagai prediksi jumlah dalam keadaan tetap, dengan cara mengalikan probabilitas keadaan tetap dengan jumlah orang yang terkait dengan perproblemaman yang sedang dihadapi.

Perproblemaman selanjutnya apabila perhitungan dilakukan per periode, tentunya akan memakan waktu yang lama, karena contohnya saja apabila probabilitas keadaan tetap ada pada periode ke-15, maka harus dilakukan perhitungan sebanyak 14 kali. Hal ini dapat diatasi dengan menghitung probabilitas keadaan tetap secara langsung menggunakan persamaan.

#### **D. Aplikasi Probabilitas Markov**

Dalam perkembangannya probabilitas markov telah diterapkan ke dalam berbagai aplikasi bidang misalnya :

1. Ilmu Fisika

Sistem markovian sudah muncul secara luas dalam termodinamika dan mekanika statistik.

2. Ilmu Kimia

Sebuah algoritma yang berdasarkan rantai Markov digunakan untuk memfokuskan pertumbuhan berdasarkan fragmen bahan kimia di silico menuju kelas yang diinginkan dari senyawa seperti obat-obatan atau produk alami.

3. Pengujian

Beberapa teoretikus telah mengusulkan gagasan tentang rantai markov uji statistik (MCST) sebuah prosedur markov rantai gabungan untuk

membentuk suatu “Markov selimut”. MCST juga memiliki kegunaan dalam keadaan dunia berbasis jaringan.

4. Teori Queueing

Rantai markov juga dapat digunakan untuk model berbagai mekanisme dalam teori queuing dan statistik

5. Ilmu Statistik

Prosedur rantai markov juga menjadi sangat penting untuk menghasilkan angka acak urutan mencerminkan secara akurat distribusi probabilitas sangat rumit yang diinginkan, melalui mekanisme yang disebut rantai Markov Monte Carlo (MCMC).

6. Ilmu Sosial

Rantai markov biasanya digunakan untuk menggambarkan argumen.

### **Contoh Probabilitas Markov**

Suatu penelitian untuk mengetahui peluang perpindahan konsumen merk kartu seluler GSM ke merk kartu seluler GSM lainnya. Penelitian ini juga akan membuat prediksi peluang perpindahan konsumen merk kartu seluler GSM di kalangan mahasiswa pada masa yang akan datang. Data yang digunakan dalam penelitian ini adalah data primer yang dikumpulkan melalui penyebaran kuesioner.

Total sampel dalam penelitian ini sebanyak 87 orang mahasiswa. Berdasarkan hasil analisis Rantai Markov pengguna GSM untuk merk AS dan Simpati yang pada awal mendominasi persentase pengguna dan bahkan pada tahun-tahun berikutnya cenderung tetap mendominasi jumlah pengguna GSM. Pada tingkat kedua, merk GSM IM3 dan GSM Tri mendominasi pengguna GSM. Masing-masing mencapai nilai antara 5-10% dari total pengguna. Selanjutnya, merk GSM XL hanya mampu

mencapai angka dibawah 5% saja. Merek GSM Axis dan Mentari belum mampu untuk masuk pada pasar. Dari kuesioner yang disebarakan, diperoleh data seperti yang tercantum pada tabel berikut :

Merek Kartu Seluler Prabayar GSM yang Digunakan

No.	Merek GSM	Jumlah Responden	Persentase (%)
1	AS	49	56.32
2	Axis	0	0
3	IM3	7	8.05
4	Mentari	0	0
5	Simpaty	20	22.98
6	Tri	8	9.20
7	XL	3	3.45
<b>Jumlah</b>		<b>87</b>	<b>100</b>

Perpindahan merek (brand switching) handphone merk GSM tertera pada tabel berikut :

Dari merek	Ke Merek								Responden Sebelumnya
	Merk GSM	AS	Axis	IM3	Mentari	Simpaty	Tri	XL	
AS	28	0	5	0	1	4	1	39	
Axis	0	0	0	0	0	0	0	0	
IM3	3	0	2	0	2	1	0	8	
Mentari	0	0	0	0	0	0	0	0	
Simpaty	2	0	0	0	15	0	0	17	
Tri	14	0	0	0	2	3	0	19	
XL	2	0	0	0	0	0	2	4	
Responden saat ini	49	0	7	0	20	8	3	87	

Dari tabel diatas dibuat tabel peluang transisi (P) yang ditunjukkan pada tabel berikut :

Dari merek	AS	IM3	Simpati	Tri	XL
AS	0.72	0.13	0.025	0.1	0.025
IM3	0.38	0.25	0.25	0.12	0
Simpati	0.12	0	0.88	0	0
Tri	0.74	0	0.1	0.16	0
XL	0.5	0	0	0	0.5
Konsumsi masing-masing merek GSM	0.56	0.08	0.23	0.1	0.03

Bila diasumsikan bahwa perpindahan merek GSM stabil, maka dapat dibuat matriks peluang transisi P sebagai berikut:

$$P = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \left[ \begin{array}{ccccc} 1 & 0.72 & 0.13 & 0.025 & 0.1 \\ 0.38 & 0.25 & 0.25 & 0.12 & 0 \\ 0.12 & 0 & 0.88 & 0 & 0 \\ 0.74 & 0 & 0.1 & 0.16 & 0 \\ 0.5 & 0 & 0 & 0 & 0.5 \end{array} \right] \end{matrix}$$

Dengan vektor periode awal :

$$X = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 & 5 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} & \left[ \begin{array}{ccccc} 0.56 & 0.08 & 0.23 & 0.1 & 0.03 \end{array} \right] \end{matrix}$$

1 = Merek GSM As

2 = Merek GSM IM3

3 = Merek GSM Simpati

4 = Merek GSM Tri

5 = Merek GSM XL

Vektor x merupakan vektor kejadian atau vektor awal diperoleh dari tabel yang dibuat dalam bentuk peluang. Untuk merek GSM AS, pada  $X_1 = 0.56$  nilai tersebut diperoleh dengan membagi jumlah konsumen merek GSM

AS dengan jumlah konsumen responden seluruhnya, yaitu  $49/87 = 0.56$  demikian seterusnya sampai dengan  $x_5 = 0.03$  diperoleh dari jumlah konsumen merek GSM XL dibagi dengan jumlah konsumen responden seluruhnya  $3/87 = 0.03$ . Prediksi jumlah konsumen kartu seluler pada periode yang akan datang ditunjukkan pada gambar berikut:

Kartu	Tahun 2012	Tahun 2013	Tahun 2014	Tahun 2015	Steady State
As	56.32	54.66	53.26	52.27	48.02
IM3	8.04	9.23	9.32	9.16	8.21
Simpati	22.98	24.71	26.38	27.82	34.23
Tri	9.19	8.23	8.06	7.90	7.07
XL	3.44	3.17	2.99	2.86	2.46

Mekanisme markov akan menuju kondisi steady state (keseimbangan) artinya setelah mekanisme berjalan beberapa periode, maka nilai matriks peluang akan selalu tetap. Pada kasus ini, berdasarkan hasil perhitungan rantai markov, diketahui bahwa nilai peluang konsumen dari masing-masing GSM akan konvergen ke satu nilai atau berada pada kondisi steady state pada transisi ke-49.

Perhitungan tersebut dilihat berdasarkan nilai-nilai peluang pada masing-masing merk GSM tetap atau konstan dan tidak akan berubah pada transisi-transisi berikutnya dengan nilai masing-masing peluang konsumen merk GSM As 48.02%; merk GSM IM3 8.21%; merk GSM Simpati 34.23%; merk GSM Tri 7.07% dan merk GSM XL 2.46%.

## E. Latihan 5

1. Jelaskan apakah definisi dari probabilitas Markov ?
2. Jelaskan kategori dari golongan Markov !
3. Dalam perkembangannya probabilitas markov telah diterapkan ke dalam berbagai aplikasi bidang, sebutkan dan jelaskan 6 pengaplikasian ilmu tentang probabilitas Markov !
4. Suatu survei di sebuah wilayah kota Jakarta diketahui bahwa wilayah tersebut terdiri dari 1000 keluarga. Dari survei tersebut, diperoleh data bahwa 600 keluarga merupakan pelanggan Toko “Jaya” dan 400 keluarga merupakan pelanggan Toko “Oye” dimana pada bulan tersebut, diketahui :
  - Dari 600 keluarga pelanggan Toko “Jaya” diperoleh data bahwa 400 keluarga tetap berbelanja di Toko “Jaya” dan 200 lainnya berbelanja di Toko “Oye”.
  - Dari 400 keluarga pelanggan Toko “Oye” dinyatakan 150 keluarga tetap berbelanja di Toko “Oye”, sedangkan 250 lainnya berbelanja di Toko “Jaya”.

Hitunglah :

- a. Matriks probabilitas transisi untuk permasalahan diatas !
- b. Probabilitas untuk Toko “Jaya” dan “Oye” pada bulan ketiga apabila pada bulan pertama keluarga tersebut memilih untuk berbelanja di Toko “Jaya”
- c. Probabilitas untuk Toko “Jaya” dan “Oye” pada bulan ketiga apabila pada bulan pertama keluarga tersebut memilih untuk berbelanja di Toko “Oye”.

- d. Nilai probabilitas pelanggan dalam keadaan tetap
- e. Jumlah perkiraan pelanggan dalam jangka panjang untuk masing-masing Toko tersebut !

# BAB 6

## Probabilitas Bayesian

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang sejarah probabilitas bayesian
  - ❖ Menjelaskan tentang Teori Bayes
  - ❖ Menjelaskan tentang
- 

### A. Sejarah Probabilitas Bayesian

Teorema Bayes ditemukan oleh Reverend Thomas Bayes tahun 1701-1761 dimana teori ini digunakan untuk menghitung nilai kebenaran probabilitas dari suatu evidence.



Gambar Reverend Thomas Bayes

Di dalam teori probabilitas dan statistika, teorema Bayes juga dikenal sebagai sebuah teori dua penafsiran yang berbeda. Dalam penafsiran Bayes, teori ini menyatakan seberapa jauh derajat kepercayaan subjektif harus berubah secara rasional ketika ada petunjuk baru. Teori ini merupakan dasar dari statistika Bayes dan memiliki pengaplikasian dalam sains, rakayasa, ilmu ekonomi, teori games, kedokteran dan hukum.

Teori probabilitas Bayes digunakan untuk menghitung probabilitas terjadinya suatu peristiwa berdasarkan pengaruh yang didapat dari pengujian. Probabilitas Bayes menerangkan hubungan antara probabilitas terjadinya hipotesis dengan terdapat fakta yang telah terjadi.

Teori probabilitas Bayes pada prinsipnya menerangkan bahwa apabila ada tambahan informasi atau evidence, maka nilai probabilitas dapat diperbaiki. Oleh karena itu, teori ini bermanfaat untuk mengubah atau memperbaiki nilai kemungkinan yang ada menjadi lebih baik dengan didukung informasi atau evidence tambahan.

## **B. Teori Bayes**

Berbagai sistem di dunia ini dipengaruhi oleh ketidakpastian karena keterbatasan sensor, gangguan lingkungan dan sebagainya. Ketidakpastian dapat didefinisikan sebagai kurangnya informasi yang memadai untuk membuat keputusan. Hal ini dapat menjadi problem, karena dapat mencegah kita untuk membuat keputusan terbaik atau bahkan dapat menyebabkan kita membuat keputusan yang buruk.

Probabilitas menyediakan cara untuk merangkum segala ketidakpastian. Ketidakpastian dapat ditangani dalam berbagai pendekatan, beberapa diantaranya

adalah : Faktor Kepastian (*Certainty Factor*), Teorema Bayes, Teorema Dempster Shafer dan Bayesian Network.

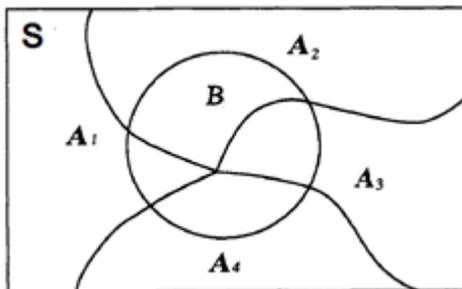
Teori probabilitas Bayes digunakan untuk menghitung probabilitas terjadinya suatu peristiwa berdasarkan pengaruh yang didapat dari pengujian. Probabilitas Bayes menerangkan hubungan antara probabilitas terjadinya hipotesis dengan fakta yang telah terjadi dan probabilitas terjadinya evidence dengan syarat hipotesis yang telah terjadi.

Teori ini didasarkan pada prinsip bahwa apabila terdapat tambahan informasi atau evidence, maka nilai probabilitas dapat diperbaiki. Oleh karena itu, teori ini bermanfaat untuk mengubah atau memperbaiki nilai kemungkinan yang ada menjadi lebih baik dengan didukung informasi atau evidence tambahan. Secara matematis persamaan teori Bayes dirumuskan :

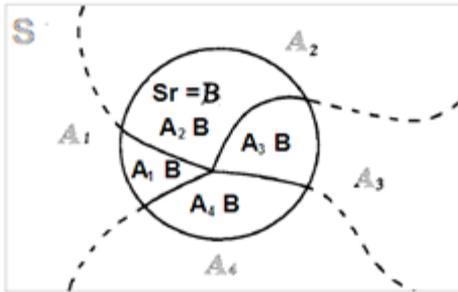
$$P(A_i | B) = \frac{P(B | A_i) P(A_i)}{\sum_j P(B | A_j) P(A_j)}$$

Lebih jelasnya lihat ilustrasi dibawah ini:

Misal  $S=A_1 + A_2 + A_3 + A_4$ , yang berarti semesta S dipartisi menjadi empat partisi, kemudian didalam S juga terdapat event B. Digambarkan sebagai berikut:



Misal event B terjadi lebih dulu, sehingga seluruh kemungkinan event di luar B menjadi tidak mungkin terjadi. Sehingga diilustrasikan sebagai berikut:



Contoh soal

Di sebuah negara, diketahui bahwa 2% dari penduduknya menderita sebuah penyakit langka. 97% dari hasil tes klinik adalah positif bahwa seseorang menderita penyakit itu. Ketika seseorang yang tidak menderita penyakit itu dites dengan tes yang sama, 9% dari hasil tes memberikan hasil positif yang salah. Apabila sembarang orang dari negara itu mengambil test dan mendapat hasil positif, berapakah peluang bahwa dia benar-benar menderita penyakit langka itu?

Jawab

$$P(A) = 2\%$$

$$P(\bar{A}) = 98\%$$

$$P(B | A) = 97\%$$

$$\begin{aligned} P(B | \bar{A}) &= 9\%P(B \cap A) \\ &= P(A) \times P(B | A) \\ &= 2\% \times 97\% = 0,0194 \end{aligned}$$

$$P(B \cap \bar{A}) = P(\bar{A}) \times P(B | \bar{A}) = 98\% \times 9\% = 0,0882$$

$$P(B \cap A) = P(A) \times P(B | A) = 2\% \times 97\% = 0,0194$$

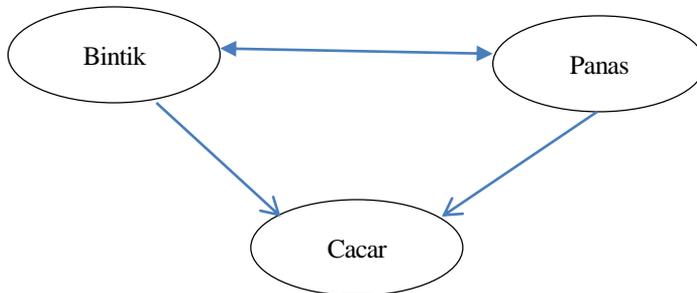
$$P(B \cap \bar{A}) = P(\bar{A}) \times P(B | \bar{A}) = 98\% \times 9\% = 0,882$$

$$\begin{aligned}
P(A | B) &= P(B \cap A) / P(B) \\
&= P(B | A) \times P(A) / P(B | A)P(A) + P(B | \bar{A})P(\bar{A}) \\
&= 97\% \times 2\% / (97\% \times 2\%) + (9\% \times 98\%) \\
&= 0.0194 / 0.0194 + 0.0882 \\
&= 0.0194 / 0.1076
\end{aligned}$$

$$P(A | B) = 0.1803$$

### C. Contoh Studi Kasus Teori Bayes

Adanya bintik-bintik di wajah merupakan gejala seseorang terkena cacar. Namun, ada observasi baru yang menunjukkan bahwa selain bintik-bintik di wajah, panas badan juga merupakan gejala orang terkena cacar. Dapat disimpulkan bahwa antara munculnya bintik-bintik di wajah dan panas badan juga memiliki keterkaitan satu sama lain.



Pada wajah Chandra terdapat bintik-bintik. Dokter menduga bahwa Chandra bila terkena cacar bila ada bintik-bintik di wajah  $\rightarrow p(\text{cacar} | \text{bintik}) = 0,8$ . Kemudian dia pasti mengalami panas badan. Diketahui probabilitas orang terkena cacar bila panas badan  $\rightarrow p(\text{cacar} | \text{panas}) = 0,5$ .

Keterkaitan antara adanya bintik-bintik di wajah dan panas badan bila seseorang terkena cacar  $\rightarrow p(\text{cacar} | \text{panas, cacar}) = 0,4$ . Sedangkan nilai keterkaitan antara adanya bintik-bintik di wajah dan panas badan  $\rightarrow p(\text{bintik} | \text{panas}) = 0,6$ .

Maka dapat ditarik kesimpulan :

$$pH|E, e = pH|E \frac{p(e|E, H)}{p(e|E)}$$

$$p \text{ cacar} | \text{panas, bintik} = p(\text{cacar} | \text{bintik}) * \frac{p(\text{bintik} | \text{panas, cacar})}{p(\text{bintik} | \text{panas})}$$

$$p \text{ cacar} | \text{panas, bintik} = 0,5 * \frac{0,4}{0,6} = 0,33$$

#### **D. Bayesian Network**

Jaringan Bayesian atau bayesian network merupakan sebuah probabilistik graphical model yang bertugas merepresentasikan sekumpulan variabel lengkap dengan nilai ketergantungan antar variabel tersebut. Secara khusus, setiap node dalam grafik merupakan variabel acak, sementara ujung-ujung antara node merupakan probabilistik dependensi acak sesuai variabel.

Dependensi kondisional dalam grafik sering diperkirakan menggunakan statistik dan dikenal sebagai komputasi prosedur. Oleh karena itu bayesian network mengabungkan prinsip-prinsip dari teori graph, teori probabilitas, komputer ilmu pengetahuan dan statistik. Bayesian network memiliki kelebihan dalam menghitung suatu probabilitas yang rumit.

Dalam pemodelannya, bayesian network menggunakan directed acyclic graph (DAG) dimana setiap node mewakili satu variabel dan panah melambangkan kondisi ketergantungan antar variabel. Misalnya apabila terdapat panah dari node A ke node B maka A dapat dikatakan sebagai parent dari B, dan sebaliknya B

adalah child dari A. Kumpulan node parent dari node X, dinotasikan sebagai parents (Xi). Peluang gabungan (*joint probability*) pada sebuah node dapat ditentukan melalui persamaan sebagai berikut :

$$p_{A_i|B} = \frac{p_{B|A_i} p_{A_i}}{\sum_j p_{B|A_j} p_{A_j}}$$

# BAB 7

## First Order Logic

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi first order logic
  - ❖ Menjelaskan tentang simbol first order logic
- 

### A. Definisi First Order Logic

Representasi dari bahasa sering disaapabilan dalam bentuk propositional logic karena cukup untuk mengilustrasikan konsep dasar dari logic. Propositional logic adalah sebuah model matematika yang memungkinkan pengguna memberikan penalaran mengenai nilai kebenaran dari sebuah ekspresi logika. Namun, propositional logic terlalu lemah untuk merepresentasikan environment yang kompleks dengan cara penyajian yang singkat.

First order logic merupakan representasi pengetahuan yang cukup ekspresif dalam menyaapabilan pengetahuan umum. First order logic adalah sistem logika formal yang digunakan dalam kajian ilmu matematika, filsafat, linguistik, dan ilmu komputer. First order logic juga dikenal sebagai orde pertama predikat kalkulus, teori kuantifikasi dan logika predikat.

Apabila proporsional logic mengasumsikan dunia berisi fakta-fakta, maka first order logic mengasumsikan bahwa dunia berisi “

Object : orang, rumah, angka, warna, permainan baseball, perang dan lain-lain.

Relation : merah, bulat, perdana, saudara, lebih besar dari, bagian dari, datang antara dan lain-lain.

Function : ayah, sahabat, satu lebih dari, ditambah dan lain-lain.

Bahasa pada first order logic dibangun dalam kaitan antara object dan relation. Hal ini tentu sangat penting di dalam matematika, ilmu filsafat, dan kecerdasan buatan karena bidang tersebut merupakan pemikiran yang sangat berguna untuk melakukan dealing antara object dan relation diantara mereka.

## B. Simbol First Order Logic

First order logic memiliki beberapa simbol yang menunjukkan maksud dari pengetahuan yang dituliskan, antara lain :

1.  $\forall$  = untuk semuanya

Contoh : setiap orang, semua orang, kapanpun, dan sebagainya. Apabila dijelaskan dalam bentuk kalimat, misalnya :

“Setiap mahasiswa di Universitas ABC adalah pintar”

Maka penggunaan universal quantifier-nya adalah :

$$\forall x: at(x, ABC) \Rightarrow Smart(x)$$

Contoh lainnya :

$$\forall x: King(x) \Rightarrow Person(x)$$

Memiliki arti : “Untuk setiap x, apabila x adalah king, maka x adalah person”.

2.  $\exists$  = ada sebagian

Contoh : seseorang, pada waktu tertentu, dsb. Apabila dijelaskan dalam bentuk kalimat, misalnya :

“Ada seseorang yang mencintai seseorang di dalam dunia ini”.

Maka penggunaan existential quantifier-nya adalah :

$\exists x: \forall y \text{ Love } (x, y)$

Contoh lainnya :

$\exists x : \text{Crown } (x) \wedge \text{onHead}(x, \text{John})$

Memiliki arti

“John memiliki sebuah crown pada kepalanya”

3.  $\Rightarrow$  implikasi

Contoh : apabila ..... maka ....

4.  $\Leftrightarrow$  ekuivalen (biconditional)

Contoh : apabila ..... dan .... hanya .... apabila ....

5.  $\neg$  = Not / negasi / ingkaran

6.  $\vee$  = Or / atau / disjungsi

7.  $\wedge$  = And / dan / konjungsi

Berikut ini beberapa contoh sederhana melakukan konversi dari bahasa natural ke

*First order logic* :

1. Muhsin adalah seorang dosen

Dosen (Muhsin)

2. Semua mahasiswa di Computer Science adalah anggota dari School of Computer Science

$\forall x: \text{mahasiswa } (x) \wedge \text{Computer Science } (x) \Rightarrow \text{anggota } (x, \text{School of Computer Science})$

3. Mahasiswa yang tidak aktif kuliah maka tidak lulus ujian

$\exists x: \text{mahasiswa } (x) \neg \text{aktif } (x, \text{kuliah}) \Rightarrow \text{lulus } (x, \text{ujian})$

4. Ada mahasiswa Universitas ABC yang memakai kaos

$\exists x$ : mahasiswa (x)  $\wedge$  memakai (x, baju)

*First order logic* lebih banyak dikaitkan ke dalam pembuktian kebenaran dari suatu fakta atau penarikan kesimpulan. Diberikan beberapa fakta atau premis, *First order logic* digunakan untuk menguji nilai kebenaran sebuah pernyataan. Pengujian ini dilakukan dengan melakukan head-to-head dari fakta / premis yang diketahui. Prosedur pengujian yang sering digunakan adalah :

1. Penalaran backward chaining

Pembuktian kebenaran suatu fakta dari beberapa premis dapat dilakukan dengan prosedur penalaran kesimpulan yang dimulai dari sasaran state lalu mundur ke belakang menggunakan fakta atau premis yang dimiliki. Contoh, misalnya ada beberapa pernyataan berikut :

- a. Lilik adalah seorang mahasiswa
- b. Lilik merupakan mahasiswa jurusan Teknik Informatika
- c. Setiap mahasiswa jurusan Teknik Informatika pasti mahasiswa Fakultas Ilmu Komputer (FIK)
- d. Kecerdasan buatan adalah mata kuliah yang mudah
- e. Setiap mahasiswa Fakultas Ilmu Komputer pasti akan menyukai mata kuliah Kecerdasan buatan atau membencinya
- f. Setiap mahasiswa pasti akan suka terhadap suatu mata kuliah
- g. Mahasiswa yang tidak pernah hadir pada mata kuliah yang mudah, maka mereka pasti tidak menyukai mata kuliah tersebut
- h. Lilik tidak pernah hadir pada mata kuliah Kecerdasan buatan

Sebelum melakukan pembuktian kebenaran atau penarikan kesimpulan terhadap suatu pernyataan yang akan diuji, maka dilakukan konversi dari kalimat tersebut ke dalam bentuk First order logic, dimana hasil konversinya yaitu:

- a. Mahasiswa (Lilik)
- b. Teknik Informatika (Lilik)
- c.  $\forall x: \text{Teknik Informatika } (x) \Rightarrow \text{FIK } (x)$
- d. Mudah (Kecerdasan buatan)
- e.  $\forall x: \text{FIK } (x) \Rightarrow \text{suka } (x, \text{ Kecerdasan buatan}) \vee \text{benci } (x, \text{Kecerdasan buatan})$
- f.  $\forall x: \exists x: \text{suka } (x,y)$
- g.  $\forall x: \exists x: \text{mahasiswa } (x) \wedge \text{mudah } (y) \wedge \neg \text{hadir } (x,y) \Rightarrow \neg \text{suka } (x,y)$
- h.  $\neg \text{hadir } (\text{Lilik, Kecerdasan Buatan})$

Setelah diperoleh bentuk First Order Logic dari semua premis atau fakta yang ada, maka dilakukan penalaran backward chaining sebagai penarikan kesimpulan. Kita ambil contoh pernyataan yang akan dicek kebenarannya :

“Apakah Lilik menyukai mata kuliah Kecerdasan Buatan?”

## 2. Re penyelesaian

### C. Contoh Studi Kasus : Analisis Semantik dengan First Order Logic

Sistem tanya jawab adalah sebuah sistem yang berupaya untuk mencari jawaban spesifik terhadap sebuah pertanyaan yang diajukan dalam bahasa natural. Dibandingkan dengan mesin penelusuran yang memberikan himpunan dokumen hasil penelusuran, sebuah sistem tanya jawab harus menghadirkan jawaban akhir kepada pengguna.

Pemanfaatan first order logic untuk merepresentasikan fakta adalah salah satu teknik dasar yang sudah lama dipakai untuk mengkodekan bahasa alami ke dalam bahasa formal. Dengan first order logic, diharapkan fakta dan juga pertanyaan dapat direpresentasikan secara tepat ke dalam konteksnya masing-masing.

First order logic dapat digunakan untuk melakukan representasi semantik yang dapat diperluas guna mengatasi berbagai permasalahan ambiguitas, misalnya untuk ambiguitas kuantifikator dapat digunakan pendekatan Montague, Cooper Storage, Keller Storage, dan Hole Semantics. Untuk mengatasi ambiguitas dalam konteks yang lebih luas, first order logic harus dilengkapi dengan konsep discourse representation structure, yang dapat dipakai untuk memilih semesta pembicaraan.

Melalui pemilihan semesta pembicaraan inilah, sebuah mekanisme tanya jawab yang merepresentasikan kebutuhan dari pengguna dapat dibangun. Keterbatasan mekanisme tanya jawab yang telah dikembangkan saat ini adalah keterbatasan dalam melakukan inferensi yang sangat tergantung pada struktur kalimat hasil analisis semantik dan sintaktik. Hal ini terjadi karena sistem masih belum dapat melakukan inferensi dari model dan hanya sebatas pada unifikasi formula first order logic antara kalimat tanya dengan semesta pembicaraan.

# BAB 8

## Logika Samar (Fuzzy Logic)

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang sejarah fuzzy logic
  - ❖ Menjelaskan tentang pengertian fuzzy logic
  - ❖ Menjelaskan tentang kelebihan fuzzy logic
  - ❖ Menjelaskan tentang aplikasi fuzzy logic
- 

### A. Sejarah Fuzzy Logic

Cikal bakal fuzzy logic sebenarnya sudah muncul pada sekitar tahun 1920-an oleh Jan Lukasiewicz ditemukannya teori kemungkinan (*probability*) yang berkisar pada sekitar angka 0 dan 1. Dilanjutkan dengan penelitian Max Black tentang ketidakjelasan (*vagueness*) dalam kajian analisis logis.

Baru kemudian *Fuzzy logic* diperkenalkan pertama kali oleh Lotfi A. Zadeh seorang profesor dan Kepala Departemen Teknik Elektrik di Universitas California USA pada tahun 1965. *Fuzzy logic* yang ditemukan oleh Zadeh memperluas teori kemungkinan menjadi sistem logika matematika formal dan konsep baru untuk mengaplikasikan istilah bahasa alami pada penelitiannya tentang *fuzzy sets*. Sistem *fuzzy logic* dapat merepresentasikan pengetahuan manusia dalam bentuk matematis dengan menyerupai cara berpikir manusia.



Gambar Lofti A. Zadeh

Fuzzy logic memiliki derajat keanggotaan dalam rentang 0 (nol) hingga 1 (satu) berbeda dengan logika digital atau diskrit yang hanya memiliki dua nilai yaitu 1 (satu) atau 0 (nol). Logika fuzzy digunakan untuk menerjemahkan suatu besaran yang diekspresikan menggunakan bahasa (linguistik). Misalnya besaran kecepatan laju kendaraan yang diekspresikan dengan pelan, agak cepat, cepat dan sangat cepat.

Fuzzy logic dapat mengolah nilai yang tidak pasti berupa batasan, seperti : sangat, sedikit, dan kurang lebih. Manusia dapat dengan mudah mengartikan kalimat “Saya pergi sebentar saja”, mungkin sebentar hanya 4 atau 5 menit. Komputer tidak mengerti nilai asli dari kata sebentar. Dengan fuzzy logic, komputer dapat mengolah ketidakpastian tersebut sehingga dapat digunakan untuk memutuskan sesuatu yang memerlukan kepintaran manusia dalam penalaran.

## **B. Pengertian Fuzzy Logic**

Secara umum fuzzy logic adalah sebuah metodologi berhitung dengan variabel kata-kata (linguistic variable), sebagai pengganti berhitung dengan bilangan. Kata-kata yang digunakan dalam fuzzy logic memang tidak sepresisi bilangan, namun

kata-kata jauh lebih dekat dengan intuisi manusia. Manusia bisa langsung merasakan nilai dari variabel kata-kata yang sudah dipakainya sehari-hari. Demikianlah, fuzzy logic memberi ruang dan bahkan mengeksploitasi toleransi terhadap ketidakpresisian. Fuzzy logic memerlukan ongkos yang lebih murah dalam memecahkan berbagai problem yang bersifat fuzzy.

Logika samar (*fuzzy logic*) adalah suatu sistem yang mengendalikan dalam pemecahan problem. *Fuzzy logic* dapat digunakan di berbagai bidang seperti : sistem diagnosis penyakit, pemodelan sistem pemasaran, riset operasi, prediksi gempa bumi, klasifikasi dan pencocokan pola.

Fuzzy logic melakukan pemetaan sebuah ruang input ke dalam ruang output dengan menggunakan If-Then rules. Pemetaan dilakukan dalam suatu *Fuzzy Inference System* (FIS). Urutan rule bisa sembarang, FIS mengevaluasi semua rule secara bersama untuk menghasilkan kesimpulan. Oleh karenanya, semua rule harus didefinisikan lebih dahulu sebelum kita membangun sebuah FIS yang akan digunakan untuk menginterpretasikan semua rule tersebut.

Untuk memahami fuzzy logic perlu diketahui adanya fuzzy set yaitu himpunan keanggotaan dari tiap elemen yang tidak mempunyai batas yang jelas. Fuzzy set merupakan pengembangan dari sebuah himpunan klasik. Apabila  $X$  adalah universe of discourse dan elemen-elemennya dinotasikan dengan  $x$ , maka sebuah fuzzy set dalam  $X$  didefinisikan dengan :

$$A = \{x, \mu_A(x) \mid x \in X\}$$

$\mu_A(x)$  adalah fungsi keanggotaan dari  $x$  dalam  $A$ . Fungsi keanggotaan memetakan tiap elemen dari  $x$  menjadi derajat keanggotaan antara 0 dan 1. Secara

umum, beberapa kesimpulan tentang himpunan dan fungsi keanggotaan fuzzy diberikan sebagai berikut :

1. Fuzzy set menekankan konsep variabel samar seperti variabel hari minggu, suhu panas, pelari cepat dan lainnya.
2. Fuzzy set mengizinkan keanggotaan parsial dari suatu himpunan seperti hari Jumat yang dianggap sebagai hari akhir minggu namun dengan derajat dibawah 1.
3. Derajat keanggotaan fuzzy dalam fuzzy set berkisar antara 0 sampai 1.
4. Tiap fungsi keanggotaan  $\mu$  berasosiasi dengan sebuah fuzzy set tertentu dan memetakan suatu nilai input ke nilai derajat keanggotaan yang sesuai. Misalnya dalam kasus fuzzy set orang berbadan tinggi mempunyai fungsi keanggotaan sendiri, yaitu  $\mu_{\text{tinggi}}$ , yang berbeda dengan fungsi keanggotaan dari fuzzy set orang berbadan rendah, yaitu  $\mu_{\text{rendah}}$ .

Dalam fuzzy logic tidak hanya variabel input yang berharga antara 0 dan 1, tetapi hasil operasi fuzzy logic-nya juga. Apabila demikian maka operasi fuzzy logic untuk AND, OR, dan NOT tidak akan sama dan sesederhana seperti dalam boolean logic.

Perbedaan operasi logika dalam boolean dan fuzzy dapat dilihat pada operasi logika AND, OR, dan NOT sebagai berikut :

Tabel

Operasi Boolean Logic

A	B	A and B	A or B	Not A	Not B
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	1
1	1	1	1	0	0

Tabel

Operasi Fuzzy Logic

A	B	Min (A,B)	Max (A,B)	1-A	1-B
0	0	0	0	1	1
0	1	0	1	1	0
1	0	0	1	0	1
1	1	1	1	0	0

### C. Kelebihan Fuzzy Logic

Kelebihan *fuzzy logic* antara lain :

1. Konsep *fuzzy logic* mudah dimengerti. Konsep matematis yang mendasari penalaran *fuzzy* sangat sederhana dan mudah dimengerti.
2. *Fuzzy logic* sangat fleksibel, dalam arti dapat dibangun dan dikembangkan dengan mudah tanpa harus memulainya dari nol.
3. *Fuzzy logic* memiliki toleransi terhadap data-data yang tidak tepat.
4. *Fuzzy logic* mampu memodelkan fungsi-fungsi non linear yang sangat kompleks

5. *Fuzzy logic* dapat membangun dan mengaplikasikan pengalaman para pakar secara langsung tanpa harus melalui training data.
6. *Fuzzy logic* dapat bekerjasama dengan teknik kendali secara konvensional yang sudah terlebih dahulu ada.
7. *Fuzzy logic* didasarkan pada bahasa alami manusia.

#### **D. Himpunan Fuzzy dan Fungsi Keanggotaan**

Himpunan fuzzy adalah kumpulan prinsip matematika sebagai penggambaran pengetahuan berdasarkan derajat keanggotaan daripada menggunakan derajat rendah dari logika biner klasik. Sebuah himpunan fuzzy adalah sebuah himpunan yang bervariasi dalam himpunan. Ini berlawanan dengan himpunan klasik karena anggota dari sebuah himpunan klasik tidak mungkin menjadi anggota, kecuali memiliki derajat keanggotaan penuh dalam himpunan. Karena tidak perlu lengkap, maka elemen-elemen tersebut juga dapat masuk menjadi anggota himpunan fuzzy yang lain pada semesta yang sama.

Tingkat keanggotaan bilangan fuzzy akan memetakan objek atau atributnya ke bilangan riil positif pada interval  $(0,1)$ . Karena karakteristik pemetaannya seperti sebuah fungsi, maka disebut sebagai fungsi keanggotaan. Fungsi keanggotaan adalah pemetaan sebuah elemen pada semesta nilai keanggotaan menggunakan sebuah fungsi teoritis.

#### **E. Aplikasi Fuzzy Logic**

Teori ini banyak digunakan di berbagai bidang seperti bidang teknologi, industri, manajemen, pertanian, transportasi, maupun medis. Berbagai contoh aplikasi fuzzy logic antara lain :

1. Pada bidang industri, fuzzy logic digunakan untuk menghasilkan service robot untuk melayani manusia.
2. Di bidang bisnis, fuzzy logic digunakan untuk memperkirakan naik turunnya harga saham atau memperkirakan keuntungan penjualan selanjutnya.
3. Pada bidang manajemen, fuzzy logic juga dimanfaatkan untuk sistem penggajian karyawan.
4. Dalam lingkungan sehari-hari, fuzzy logic juga banyak ditemukan pada mesin cuci dan pemanas ruangan.
5. Fuzzy logic juga telah masuk dalam bidang pertanian, misalnya untuk meramal cuaca sebelum para petani mulai menanam.

#### **F. Contoh Studi Kasus Fuzzy Logic**

Dalam software Matlab, pengaplikasian *Fuzzy Logic* dapat dilihat dari *Fuzzy Logic Toolbox* yaitu sekumpulan tool yang akan membantu merancang sistem fuzzy untuk diaplikasikan dalam berbagai bidang, seperti : *automatic control, signal processing, identification system, pattern recognition, time series prediction, data mining*, dan bahkan *financial applications*.

Salah satu tool dalam *Fuzzy Logic Toolbox* yang dapat digunakan untuk aplikasi Fuzzy Logic yaitu menggunakan *Graphical User Interface* (GUI) yang terdiri dari 5 jenis meliputi : *FIS Editor, Membership Function Editor, Rule Editor, Rule Viewer*, dan *Surface Viewer*.

Contoh kasus yaitu dalam penentuan bonus untuk seorang pelayan di restoran. Besar bonus ditentukan dari dua standarisasi, kualitas pelayanan dan kualitas makanan. Ketika seorang pelayan melayani pelanggannya, pelanggan bisa dengan

cepat menilai seberapa bagus pelayanan dari pelayan dan seberapa enak makanan yang disaapabilan, dan kemudian dengan mudah pula mengambil keputusan berapa besar bonus yang seharusnya diberikan kepada pelayan tersebut. Kedua variabel tersebut, yaitu pelayanan dan kualitas makanan, jelas bersifat fuzzy.

Dari kasus tersebut muncul pertanyaan bagaimana cara supaya kemampuan yang dimiliki manusia ini bisa dilakukan oleh sebuah mesin. Hal ini dapat dilakukan dengan menggunakan FIS (*Fuzzy Inference System*) Langkah pertama dengan meng-konversi variabel input yaitu pelayanan dan makanan ke dalam rentang bilangan / angka, sehingga bisa dimengerti oleh FIS. Skor yang digunakan berkisar antara 0 sampai dengan 10. Untuk kualitas pelayanan skor 0 berarti mengecewakan dan skor 10 berarti memuaskan. Sedangkan untuk kualitas makanan skor 0 berarti tidak lezat dan skor 10 berarti lezat. Sedangkan variabel output-nya adalah bonus yang diterima pelayan. Besarnya bonus menggunakan skor 0 yaitu sedikit dan skor 10 yaitu banyak.

Selanjutnya rule untuk menyatakan besarnya bonus yang diterima pelanggan dengan mempertimbangkan kualitas pelayanan dan kualitas makanan di restoran dapat dilihat berikut :

If pelayanan is mengecewakan Or makanan is hambar

Then bonus is sedikit

If pelayanan is bagus

Then bonus is sedang

If pelayanan is memuaskan Or makanan is lezat

Then bonus is banyak

# BAB 9

## **Pengolahan Bahasa Alami (*Natural Language Processing*)**

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi natural language processing
  - ❖ Menjelaskan tentang tingkat pengolahan bahasa alami
  - ❖ Menjelaskan tentang area penelitian bahasa alami
  - ❖ Menjelaskan tentang terminologi pengolahan bahasa alami
  - ❖ Menjelaskan tentang information retrieval
  - ❖ Menjelaskan tentang aplikasi pengolahan bahasa alami
  - ❖ Menjelaskan tentang semantic
- 

### **A. Definisi Natural Language Processing**

*Natural language processing* adalah pemrosesan bahasa alami sehingga user dapat berkomunikasi dengan komputer. *Natural language processing* adalah pembuatan program yang memiliki kemampuan dalam memahami bahasa manusia. *Natural language processing* berfokus pada interaksi antara komputer dengan bahasa alami yang digunakan manusia. Mekanisme komputer melakukan ekstraksi informasi dari input yang berupa bahasa alami dan menghasilkan output berupa natural language.

Tujuan dari pengolahan bahasa alami yaitu melakukan mekanisme pembuatan model komputasi dari bahasa, sehingga dapat terjadi suatu interaksi antara manusia dengan komputer melalui perantara bahasa alami. Model komputasi ini dapat

berguna untuk keperluan ilmiah misalnya meneliti sifat-sifat dari suatu bentuk alami maupun untuk keperluan sehari-hari dalam hal ini memudahkan komunikasi antara manusia dengan komputer.

Pada pengolahan bahasa alami harus memperhatikan pengetahuan terhadap bahasa itu sendiri, baik dari segi kata yang digunakan, bagaimana kata-kata tersebut digabung untuk menghasilkan suatu kalimat, apa arti sebuah kata, apa fungsi sebuah kata dalam sebuah kalimat dan sebagainya. Akan tetapi kita juga harus mempertimbangkan ada satu hal lagi yang sangat berperan dalam bahasa yaitu kemampuan manusia untuk mengerti dan kemampuan untuk itu didapat dari pengetahuan yang didapat secara terus menerus sewaktu hidup. Sebagai contoh dalam suatu percakapan, seseorang mungkin dapat menjawab suatu pertanyaan atau ikut dalam suatu percakapan dengan tidak hanya berdasar pada kemampuan berbahasa, tetapi juga harus tahu misalnya kata istilah yang umum digunakan dalam kelompok percakapan itu juga harus tahu misalnya kata istilah yang umum digunakan dalam kelompok percakapan itu atau bahkan harus tahu konteks dari percakapan tersebut.

## **B. Tingkat Pengolahan Bahasa Alami**

Tingkat pengolahan bahasa alami (*natural language processing*) meliputi :

1. Fonetik dan fonologi : berhubungan dengan suara yang menghasilkan kata yang dapat dikenali. Bidang ini menjadi penting dalam mekanisme aplikasi yang memakai prosedur *speech based system*.
2. Morfologi : yaitu pengetahuan tentang kata dan bentuknya dimanfaatkan untuk membedakan satu kata dengan lainnya. Pada tingkat ini juga dapat dipisahkan antara kata dengan elemen lain seperti tanda baca.

3. Sintaksis : yaitu pemahaman tentang urutan kata dalam pembentukan kalimat dan hubungan antara kata tersebut dalam mekanisme perubahan bentuk dari kalimat menjadi bentuk yang sistematis, meliputi mekanisme pengaturan tata letak suatu kata dalam kalimat akan membentuk kalimat yang dapat dikenali. Selain itu dapat pula dikenali bagian-bagian kalimat dalam suatu kalimat yang lebih besar.
4. Semantik : yaitu pemetaan bentuk struktur sintaksis dengan memanfaatkan tiap kata ke dalam bentuk yang lebih mendasar dan tidak tergantung struktur kalimat. Semantik mempelajari arti suatu kata dan bagaimana dari arti kata tersebut membentuk suatu arti dari kalimat yang utuh. Dalam tingkatan ini belum tercakup konteks dari kalimat tersebut.
5. Pragmatik : pengetahuan pada tingkatan ini berkaitan dengan masing-masing konteks yang berbeda tergantung pada situasi dan tujuan pembuatan sistem.
6. Discourse knowledge : melakukan pengenalan apakah suatu kalimat yang sudah dibaca dan dikenali sebelumnya akan mempengaruhi arti dari kalimat selanjutnya.
7. World knowledge : mencakup arti sebuah kata secara umum dan apakah ada arti khusus bagi suatu kata dalam suatu percakapan dengan konteks tertentu.

### **C. Area Penelitian Bahasa Alami (NLP)**

Ada beberapa area utama penelitian pada bidang Pemrosesan Bahasa Alami :

1. *Question Answering System* (QAS)

QAS adalah kemampuan komputer untuk menjawab pertanyaan yang diberikan oleh user. Daripada memasukkan keyword ke dalam browser penelusuran, dengan QAS, user dapat langsung bertanya dalam bahasa alami yang digunakannya, baik Bahasa Inggris, Bahasa Mandarin, atau Bahasa Indonesia.

2. *Summarization*

Pembuatan ringkasan dari sekumpulan konten dokumen atau email. Aplikasi membantu user mengkonversikan dokumen teks yang besar ke dalam bentuk slide presentasi.

3. *Machine Translation*

Produk yang dihasilkan adalah aplikasi yang dapat memahami bahasa manusia dan menerjemahkannya ke bahasa lain. Termasuk di dalamnya adalah google translate yang apabila dicermati semakin membaik dalam penerjemahan bahasa.

4. *Speech Recognition*

Bagian ini merupakan cabang dari ilmu bahasa alami yang cukup sulit. Mekanisme pembangunan model untuk digunakan telepon atau komputer dalam mengenali bahasa yang diucapkan sudah banyak dikerjakan. Bahasa yang sering digunakan berupa pertanyaan dan perintah.

5. *Document Classification*

Aplikasi ini merupakan area penelitian bahasa alami (NLP) yang paling sukses. Pekerjaan yang dilakukan aplikasi ini adalah menentukan dimana tempat terbaik dokumen yang baru diinputkan ke dalam sistem. Hal ini sangat berguna pada aplikasi spam filtering, news article classification dan movie review.

#### **D. Terminologi NLP**

Perkembangan Bahasa Alami (NLP) menghasilkan kemungkinan dari interface bahasa alami menjadi knowledge base dan penerjemahan bahasa alami.

Ada 3 aspek utama pada teori pemahaman mengenai bahasa alami antara lain :

1. Syntax

Syntax menjelaskan bentuk dari bahasa, biasa dispesifikasikan oleh sebuah grammar. Bahasa alami melebihi formal language yang digunakan untuk logika kecerdasan buatan dan program komputer.

2. Semantics

Semantics menjelaskan arti dari kalimat dalam satu bahasa, meskipun teori semantic secara umum telah ada, ketika membangun sistem bahasa alami untuk aplikasi tertentu, akan digunakan representasi yang paling sederhana.

3. Pragmatics

Komponen pragmatics menjelaskan hubungan pernyataan yang ada dengan dunia. Untuk memahami bahasa, agen harus mempertimbangkan lebih lanjut, tidak hanya sekadar kalimat. Agen harus melihat lebih ke dalam konteks kalimat, keadaan dunia, tujuan dari speaker, listener, konvensi khusus dan sejenisnya.

Selain ketiga istilah diatas, terdapat beberapa istilah yang terkait dengan Bahasa Alami (NLP) yaitu :

1. Morfologi, adalah pengetahuan tentang kata dan bentuknya sehingga dapat dibedakan antara yang satu dengan yang lainnya. Dapat juga didefinisikan asal usul sebuah kata. Contoh : membangun : bangun (kata dasar), mem- (prefix), -kan (suffix).

2. Fonetik, adalah segala hal yang berhubungan dengan suara yang menghasilkan kata yang dapat dikenali. Fonetik digunakan dalam pengembangan NLP khususnya bidang speech based system.

### **E. Information Retrieval**

*Information retrieval* (IR) adalah pekerjaan menemukan dokumen yang relevan dengan kebutuhan informasi user. Contoh sistem IR yang paling populer adalah search engine pada world wide web. Seorang pengguna web dapat menginputkan query berupa kata apapun ke dalam sebuah search engine dan melihat hasil dari penelusuran yang relevan. Karakteristik dari sebuah sistem IR adalah :

1. A corpus of documents  
Setiap sistem harus memutuskan dokumen yang ada akan diperlakukan sebagai apa. Bisa sebagai sebuah paragraf, halaman, atau teks multipage.
2. Queries posed in a query language  
Setiap sistem harus memutuskan dokumen yang ada akan diperlakukan sebagai apa. Bisa sebagai sebuah paragraf, halaman, atau teks multipage.
3. A result set  
Ini adalah bagian dari dokumen yang dinilai oleh sistem IT sebagai yang relevan dengan query
4. A presentation of the result set  
Maksud dari bagian ini adalah tampilan list judul dokumen yang sudah dirangking.

### **F. Aplikasi Pengolahan Bahasa Alami**

Beberapa aplikasi dalam bidang pengolahan bahasa alami (natural language processing) meliputi : text based application dan dialogue based application. *Text based application* mencakup segala macam aplikasi yang melakukan mekanisme terhadap teks tertulis seperti misalnya : buku, berita di surat kabar, e-mail dan lainnya. Contoh penggunaan dari *text based application* ini adalah :

1. Mencari topik tertentu dari buku yang ada pada perpustakaan
2. Mencari isi dari surat atau e-mail
3. Menterjemahkan dokumen dari satu bahasa ke bahasa yang lain

*Dialogue based application* dimana pendekatan ini melibatkan bahasa lisan atau pengenalan suara, akan tetapi bidang ini juga memasukkan interaksi dengan cara memasukkan teks pertanyaan melalui keyboard. Aplikasi yang sering ditemui untuk bidang ini adalah :

1. Sistem tanya jawab, dimana pengolahan bahasa alami digunakan dalam mendapatkan informasi dari suatu database.
2. Sistem otomatis pelayanan melalui telepon
3. Control suara pada peralatan elektronik
4. Sistem problem solving yang membantu untuk melakukan pemecahan problem yang umum dihadapi dalam suatu pekerjaan

## **G. Semantic**

*Semantic analyzer* mempunyai hubungan rule dalam basis pengetahuan untuk menginterpretasikan sebuah kalimat.

Rule 1 :

If determiner adalah bagian pertama dalam kalimat dan diikuti oleh noun then noun tersebut dianggap sebagai subyek

Rule 2 :

If verb diikuti subyek then verb menjelaskan tentang apa yang dikerjakan oleh subyek

Rule 3 :

If noun diikuti subyek dan verb then noun tersebut dianggap sebagai obyek

Rule 4 :

If kalimat mempunyai bentuk subyek, ver, obyek then subyek mengerjakan (verb) yang ada hubungannya dengan obyek.

## **H. Contoh Kasus : Pendeteksian Kesamaan Dokumen**

Dalam hal ini dapat dilakukan dengan menciptakan perangkat lunak yang dapat memeriksa kesamaan dari dua atau lebih dokumen. Nilai perbandingan kemiripan antara dokumen akan dihitung untuk menghasilkan prosentase toleransi kemiripan. Dengan nilai perbandingan tersebut, diharapkan dapat membantu pengguna dalam mendeteksi kesamaan kata pada dokumen secara tepat, cepat dan otomatis. Sebuah dokumen dikatakan telah menjiplak dokumen lain apabila memiliki tingkat kesamaan yang melebihi batas toleransi tertentu yang telah ditentukan.

Dalam menghitung nilai toleransi, algoritma Winnowing dan prosedur Gram dapat digunakan dalam membangun aplikasi tersebut. Winnowing menggunakan window sebagai prosedurnya, yaitu pembentukan window setelah mekanisme

hasiing. K-Gram adalah rangkaian terms dengan panjang K. Terms yang dimaksud adalah sebuah kata. Prosedur K-Gram digunakan untuk mengambil potongan-potongan karakter sejumlah K dari semua kata yang terdapat pada dokumen sumber.

Urutan metodologi aplikasi pendeteksi kesamaan dokumen terdapat 5 langkah sebagai berikut :

1. Filtering

Penghapusan karakter-karakter pada dokumen yang tidak relevan. Contoh dari karakter ini adalah spasi dari tanda baca (contoh : titik, koma, tanda seru dan sebagainya)

2. K-Gram

Pembentukan rangkaian gram dengan ukuran K. Nilai K di-input oleh pengguna untuk menentukan batas toleransi kesamaan kata yang terdapat dalam dokumen. Dengan asumsi 1 kata sama dengan 5 K-Gram, pengguna dapat mengatur ukuran menjadi 25 K-Gram apabila pengguna ingin batas toleransi sebanyak 5 kata. Dengan begini, bagian kata (substring) baru akan sama apabila mereka memiliki 5 atau lebih kata yang sama.

3. Hashing

Setelah rangkaian gram telah dibentuk, perhitungan nilai hash dari setiap gram dilakukan. Dengan hashing, rangkaian gram diubah menjadi nilai atau kode yang menjadi penanda dari rangkaian gram tersebut.

4. Window

Nilai-nilai hash tersebut dibagi ke dalam window tersebut.

## 5. Finger print

Setelah pembentukan window yang berisi nilai-nilai hash, maka dipilih nilai hash yang paling kecil dari setiap window. Apabila ada lebih dari satu nilai yang paling kecil dimekanisme windowing, maka nilai yang terkecil pada window sebelumnya itu dipilih untuk dijadikan document's fingerprints sebagai dasar pembandingan antar dokumen.

### **I. Contoh Kasus : Aplikasi Peringkas Dokumen**

Information overload menjadi salah satu dampak negatif dari kemudahan mekanisme manipulasi, penyimpangan, dan distribusi informasi digital. Besarnya jumlah dokumen teks yang tersedia secara digital mengakibatkan terlalu banyak informasi yang harus diakses dan dianalisis oleh pengguna dengan cepat. Pengguna harus dapat mengambil intisari dari dokumen teks untuk membantu pengambilan keputusan berdasarkan informasi yang tersedia. Automatic text summarization adalah salah satu penyelesaian untuk mengatasi problem information overload tersebut.

# BAB 10

## Game Playing

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi game playing
  - ❖ Menjelaskan tentang jenis-jenis game playing
  - ❖ Menjelaskan tentang
- 

### A. Definisi Game Playing

*Game* adalah fasilitas yang sangat menarik dalam komputer. Ide games pertama kali dimunculkan oleh Claude Shannon pada tahun 1950 dan dilanjutkan dengan Alan Turing yang menemukan program permainan catur. Kemudian pada tahun 1960-an Arthur Samuel mencoba membuat program catur tersebut. Beberapa alasan mengapa games merupakan domain yang baik untuk dikembangkan yaitu :

1. Sangat mudah untuk menentukan ukuran kesuksesan dan kegagalannya (menang atau kalah)
2. Tidak memerlukan terlalu banyak pengetahuan. Permainan dapat diselesaikan dengan melakukan penelusuran dari arah start sampai posisi menang.
3. Ruang keadaannya mudah direpresentasikan
4. Operator-operator yang digunakan tidak terlalu banyak
5. Sebagian besar game dapat dimodelkan dengan mudah
6. Sangat mungkin untuk dibandingkan dengan kemampuan manusia.

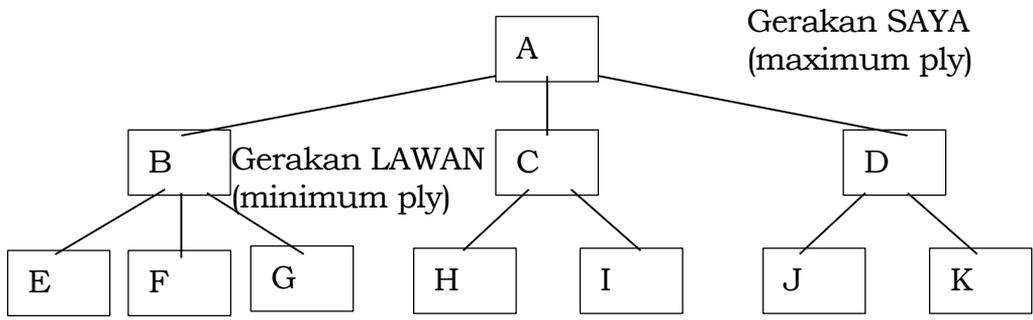
*Game playing* adalah game yang diwakili oleh pohon-pohon dimana node-node menunjukkan semua kemungkinan keadaan game dan sisi-sisi (edges) mewakili langkah antara kedua pemain. Game playing adalah problem penelusuran yang meliputi komponen-komponen berikut ini :

1. Keadaan awal (initial state), yaitu keadaan yang mendefinisikan konfigurasi awal permainan dan
2. mengidentifikasi pemain pertama yang bergerak.
3. Fungsi penerus (*successor function*), yang bertugas mengidentifikasi kemungkinan-kemungkinan yang dapat dicapai dari keadaan saat ini. Fungsi ini membuat sebuah daftar pasangan gerak dan keadaan, yang masing-masing pasangan menunjukkan langkah legal dan keadaan yang dihasilkan.
4. Sasaran test, yang bertugas untuk memeriksa apakah suatu keadaan tertentu adalah keadaan tujuan atau bukan. Keadaan-keadaan dimana permainan berakhir disebut sebagai keadaan terminal.
5. *Path cost / utility / payoff function*, yang memberikan nilai numerik untuk keadaan-keadaan terminal.

## **B. Jenis-jenis Game Playing**

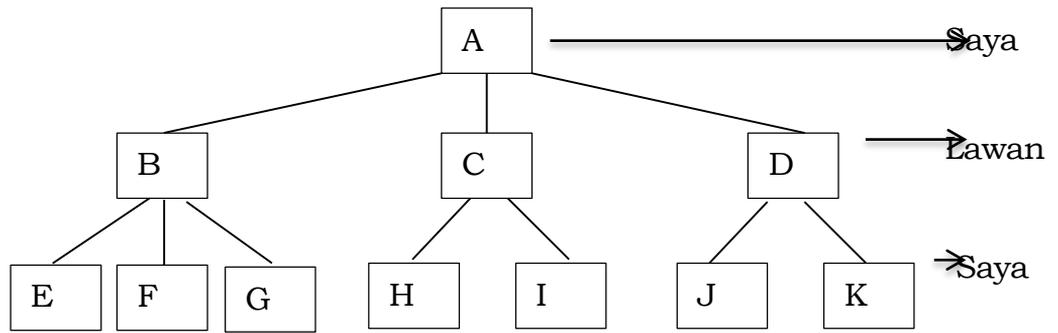
### **1. Minimax**

Salah satu teknik games yang terkenal adalah minimax dengan menggunakan teknik *depth first search* dengan kedalaman terbatas. Fungsi evaluasi yang digunakan adalah fungsi evaluasi statis, dengan mengasumsikan bahwa lawan akan membuat langkah terbaik yang mungkin.



Gambar Satu Ply Search

Pada minimax kita mengenal adanya istilah ply yaitu gerakan saya dan lawan. Games dengan 1 ply search dan 2 ply search.



Gambar Dua Ply Search

Algoritma Minimax :

Minimax (StatusSaya, Kedalaman, Pemain)

If (Kedalaman=Max)

    Return static(StatusSaya, Pemain)

Bangkitkan successor S[1..n]

If (Pemain = Saya)

    Return max of Minimax (S[i], kedalaman+1, Lawan)

Else

    Return min of Minimax (S[i], kedalaman+1, Saya)

Pada Minimax ada 2 prosedur yang dijalankan, yaitu Maksimisasi dilakukan oleh Saya dan Minimisasi dilakukan oleh Lawan. Dalam hal ini diperlukan suatu fungsi evaluasi statis yang menyatakan nilai yang mungkin didapat oleh pemain, misalkan nilai tersebut antara -10 sampai +10.

# BAB 11

## Algoritma Genetika

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi algoritma genetika
- ❖ Menjelaskan tentang struktur utama algoritma genetika
- ❖ Menjelaskan tentang

### A. Definisi Algoritma Genetika

Algoritma genetika adalah teknik penelusuran heuristik yang didasarkan pada evolusi seleksi alam dan genetik. Algoritma ini memanfaatkan mekanisme seleksi alamiah yang dikenal dengan mekanisme evolusi. Dalam mekanisme evolusi, individu secara terus menerus mengalami perubahan gen untuk menyesuaikan dengan lingkungan hidupnya. Hanya individu yang kuat yang mampu bertahan. Mekanisme seleksi alam ini melibatkan perubahan gen yang terjadi pada individu melalui mekanisme perkembangbiakan.

Algoritma genetik merupakan algoritma penelusuran yang berdasarkan pada mekanisme sistem natural yakni genetik dan seleksi alam. Dalam aplikasi algoritma genetik, variabel penyelesaian dikodekan ke dalam struktur string yang merepresentasikan barisan gen, yang merupakan karakteristik dari penyelesaian problem.

Berbeda dengan teknik penelusuran konvensional, algoritma genetik berangkat dari himpunan penyelesaian yang dihasilkan secara acak. Himpunan ini disebut populasi. Sedangkan setiap individu dalam populasi disebut kromosom yang

merupakan representasi dari penyelesaian. Kromosom-kromosom berevolusi dalam suatu mekanisme iterasi yang berkelanjutan yang disebut generasi. Pada setiap generasi, kromosom dievaluasi berdasarkan suatu fungsi evaluasi (Gen dan Cheng,1997).

Mekanisme perkembangbiakan ini didasarkan pada analogi struktur genetik dan perilaku kromosom dalam populasi individu dengan menggunakan dasar sebagai berikut :

1. Individu dalam populasi bersaing untuk sumber daya alam dan pasangannya.
2. Mereka yang paling sukses di setiap kompetisi akan menghasilkan keturunan yang lebih baik daripada individu yang berkinerja buruk
3. Gen dari individu yang baik akan menyebar ke seluruh populasi sehingga dua orangtua yang baik, kadang-kadang akan menghasilkan keturunan yang lebih baik dari orangtuanya.
4. Setiap ada pergantian generasi maka generasi terbaru ini biasanya lebih cocok dengan lingkungan mereka. Dengan kata lain, generasi baru ini bisa menyesuaikan dengan keadaan lingkungannya.

Algoritma genetika adalah algoritma penelusuran heuristik yang didasarkan atas mekanisme evolusi biologis. Keberagaman pada evolusi biologis adalah variasi dari kromosom antar individu organisme. Variasi kromosom ini akan mempengaruhi laju reproduksi dan tingkat kemampuan organisme untuk tetap hidup. Pada dasarnya ada 4 kondisi yang sangat mempengaruhi mekanisme evaluasi yaitu :

1. Kemampuan organisme untuk melakukan reproduksi

2. Keberadaan populasi organisme yang bisa melakukan reproduksi
3. Keberagaman organisme dalam suatu populasi
4. Perbedaan kemampuan untuk survive

Individu yang lebih kuat (fit) akan memiliki tingkat survival dan tingkat reproduksi yang lebih tinggi apabila dibandingkan dengan individu yang kurang fit. Pada kurun waktu tertentu (sering dikenal dengan istilah generasi), populasi secara keseluruhan akan lebih banyak memuat organisme yang fit.

Algoritma genetika pertama kali dikembangkan oleh John Holland dari Universitas Michigan pada tahun 1975, dia menyatakan bahwa setiap problem yang berbentuk adaptasi (alami dan buatan) dapat diformulasikan dalam terminologi genetika. Algoritma genetika adalah simulasi dari mekanisme evolusi Darwin dan operasi genetika tas kromosom.

## **B. Struktur Umum Algoritma Genetika**

Pada algoritma ini, teknik penelusuran dilakukan sekaligus atas jumlah penyelesaian yang dikenal dengan populasi. Individu yang terdapat dalam satu populasi disebut dengan istilah kromosom. Kromosom ini merupakan suatu penyelesaian yang masih berbentuk simbol. Populasi awal dibangun secara acak, sedangkan populasi berikutnya merupakan hasil evaluasi dengan menggunakan alat ukur yang disebut dengan fungsi fitness. Nilai fitness dari suatu kromosom akan menunjukkan kualitas kromosom dalam populasi tersebut. Generasi berikutnya dikenal dengan istilah anak (*offspring*) terbentuk dari 2 kromosom generasi sekarang yang bertindak sebagai induk (*parent*) dengan menggunakan operator penyilangan (*crossover*). Selain operator penyilangan, suatu kromosom dapat juga

dimodifikasi dengan menggunakan operator mutasi. Populasi generasi yang baru dibentuk dengan cara menyeleksi nilai fitness dari kromosom induk (*parent*) dan nilai fitness dari kromosom anak (*offspring*), serta menolak kromosom lainnya sehingga ukuran populasi (jumlah kromosom dalam suatu populasi) konstan. Setelah melalui beberapa generasi, maka algoritma ini akan konvergen ke kromosom terbaik.

# BAB 12

## Computer Vision

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang sekilas computer vision
  - ❖ Menjelaskan tentang sejarah computer vision
  - ❖ Menjelaskan tentang aplikasi computer vision
- 

### A. Sekilas Computer Vision

Manusia menerima struktur gambar tiga dimensi dari dunia disekitarnya dengan mudah, misalnya ketika melihat sebuah pot bunga yang terletak di meja. Kita dapat bercerita dengan baik mengenai bentuk dan ketebalan dari setiap daun bunga dengan melibatkan aspek pencahayaan dan bayangan dari daun tersebut. Kita juga dapat melihatnya dari sudut permukaan dan segmen setiap daun berdasarkan *background*-nya.

*Computer vision* merupakan teknologi yang memainkan peranan penting dalam pengembangan berbagai aplikasi cerdas di masa mendatang. *Computer vision* merupakan ilmu pengetahuan dari mesin untuk mendapatkan informasi berupa image, video sequence, view dari multiple camera, atau data multidimensi dari sebuah scanner.

Apabila dirumuskan maka computer vision adalah *camera + computer + pattern recognition* (mekanisme pengolahan dari data menjadi informasi). *Pattern recognition* merupakan bagian dari *computer vision*.

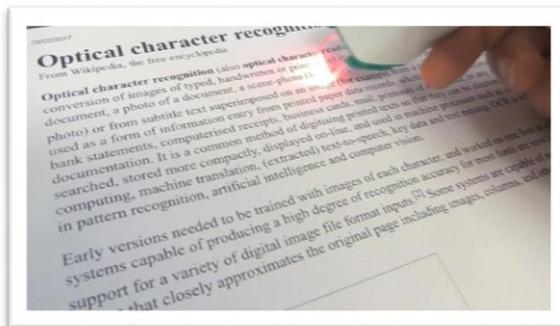
## B. Sejarah Computer Vision

*Computer vision* merupakan teknologi baru yang menarik untuk terus dipelajari secara mendalam dan berkembang sejak tahun 1966. Pada awalnya tahun 1966 Minsky membuat *computer vision* sebagai proyek kuliah pada musim panas. Pada tahun 1970-an beberapa progress dalam menginterpretasikan image yang diseleksi. Pada tahun 1980-an pergeseran ke arah geometri dan peningkatan mathematical rigor. Pada tahun 1990-an pengenalan wajah, analisis statistik pada mode. Pada tahun 2000-an pengenalan lebih luas, dataset beranotasi besar, inisiasi video processing.

## C. Aplikasi Computer Vision

Beberapa aplikasi dari *Computer vision* antara lain :

1. *Optical Character Recognition*, contoh pembacaan kode pos yang ditulis tangan pada surat dan pengenalan nomor plat secara otomatis (*Automatic Number Plate Recognition / ANPR*)



2. *Machine Inspection*, contoh : inspeksi bagian yang cepat untuk jaminan kualitas saat menggunakan stereo vision dengan iluminasi khusus saat memperkirakan toletansi pada sayap pesawat, bagian tubuh, atau mencari kerusakan dalam pengecoran baja menggunakan X-ray.



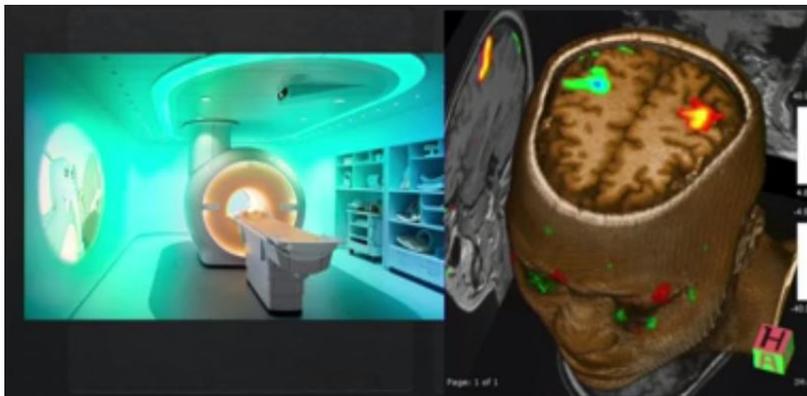
3. *Retail*, contoh : pengenalan objek untuk jalur checkout otomatis.



4. *3D Model Building (Photogrammetry)*, contoh : konstruksi otomatis dari model 3D dari foto udara pada sistem seperti pada Bing Maps.



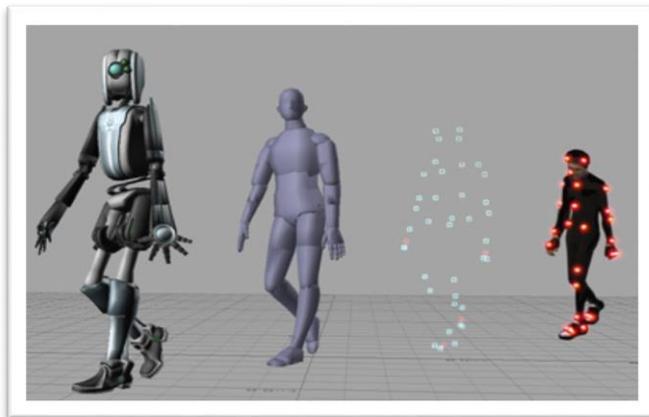
5. *Medical Imaging*, contoh : penyimpanan image sebelum dan sesudah operasi atau dalam melakukan studi jangka panjang mengenai morfologi otak manusia seiring bertambahnya usia.



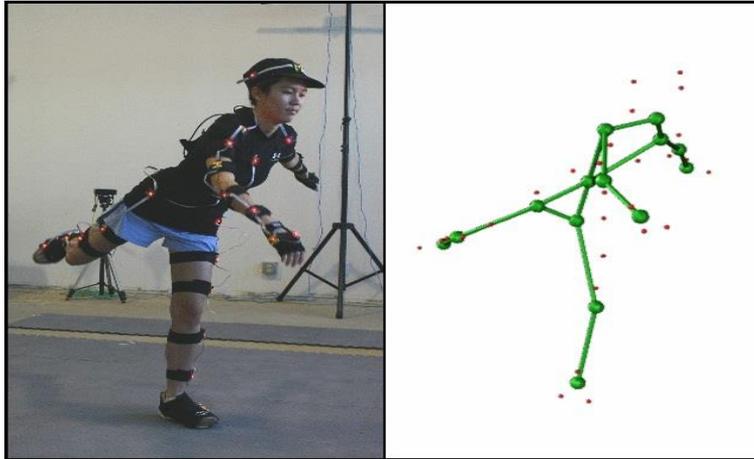
6. *Automotive Safety*, contoh : pendekatan halangan yang tidak terduga, misalnya menghindari pejalan kaki. Dapat juga digunakan saat kondisi teknik penglihatan aktif seperti radar yang tidak bekerja dengan baik.



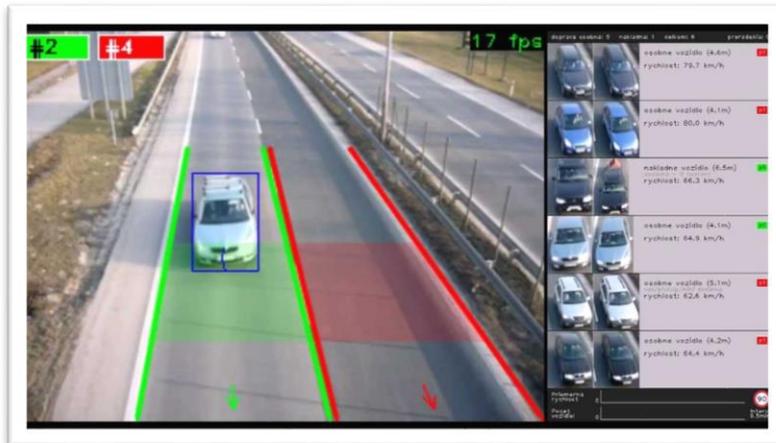
7. *Match move*, contoh : penggabungan *computer generated imagery* (CGI) dengan rekaman *live action* saat melakukan *tracking feature point* pada *environment*.



8. *Motion capture*, contoh : menggunakan penanda *retro – reflective* yang diperlihatkan melalui *multiple camera* atau teknik berbasiskan vision untuk memperoleh aktor dari animasi komputer.



9. *Surveillance*, contoh : monitoring untuk pengacau, analisis lalu lintas yang tinggi, dan monitoring kolam untuk pencegahan orang tenggelam.



10. *Fingerprint Recognition and Biometrics*, contoh : untuk akses autentifikasi otomatis seperti biasa digunakan pada aplikasi forensik.



# BAB 13

## Robotika

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang definisi robotika
  - ❖ Menjelaskan tentang kategori robotika
  - ❖ Menjelaskan tentang robot hardware
  - ❖ Menjelaskan tentang arsitektur software robot
  - ❖ Menjelaskan tentang aplikasi robot
- 

### A. Definisi Robotika

Bidang ilmu robotika mempelajari mekanisme merancang robot yang berguna bagi industri yang fungsinya membantu dan menggantikan tugas manusia. Robot mampu melakukan beberapa tugas melalui interaksi dengan lingkungan sekitar. Untuk melakukan hal tersebut, robot diperlengkapi dengan actuator seperti lengan, roda, kaki dan lainnya.

Robot juga dilengkapi dengan sensor untuk menerima dan bereaksi terhadap lingkungan. Saat ini, robot menggunakan beragam kumpulan sensor, termasuk kamera serta laser untuk pengukuran lingkungan, giroskop, serta accelerometer untuk pengukuran gerakan robot itu.

Robot berbasis pada dua enabling technology yaitu telemanipulator dan kemampuan numeric control mesin. Telemanipulator mesin yang dikendalikan dengan remote biasanya terdiri dari lengan dan gripper. Pergerakan dari lengan dan gripper mengikuti instruksi yang diberikan melalui *control device*.

Telem manipulator pertama yang digunakan untuk materi radioaktif, numeric control memungkinkan mengendalikan mesin dengan tepat berkaitan dengan sistem koordinat yang diberikan. Telem manipulator yang pertama kali digunakan yaitu tahun 1952 yang menghasilkan bahasa pemrograman pertama untuk mesin yaitu Automatic Programmed Tools (APT). Kombinasi dari kedua teknik ini berbuah pada telem manipulator yang dapat diprogram untuk pertama kalinya. Industri robot pertama yang menggunakan prinsip ini diinstalasi pada tahun 1961.

Kata robot diperkenalkan ke publik oleh seorang penulis dari Cekoslovakia yaitu Karel Capek sekitar tahun 1920-an. Istilah robotic diawali oleh Isaac Asimov pada tahun 1941 pada cerita pendeknya yang bernuansa science fiction dengan judul “Liar”.

## **B. Kategori Robotika**

Robot pada umumnya terbagi menjadi tiga kategori, yaitu manipulator robot, mobile robot dan mobile manipulator.

### **1. Manipulator robot**

Secara fisik, lengan manipulator robot terhubung dengan workplace. Misalnya, dalam sebuah jalur perakitan pada pabrik. Gerakan manipulator biasanya mengikutsertakan rantai yang dikontrol oleh persendiannya. Hal ini memungkinkan setiap robot untuk menempatkan effector-nya pada posisi dimanapun di workplace-nya.

Manipulator merupakan tipe yang paling umum dalam industri robot. Jumlah manipulator robot yang terinstalasi di dunia sekitar satu juta unit. Beberapa mobile manipulator digunakan di rumah sakit untuk membantu mekanisme operasi. Produksi mobil juga banyak melibatkan mobile

manipulator dan beberapa manipulator tersebut digunakan untuk mengenerate karya asli mereka.

## 2. Mobile robot

Jenis robot ini bergerak sesuai lingkungan menggunakan roda, lengan, atau mekanisme lainnya. Mobile robot biasanya ditugaskan untuk mengirimkan makanan di rumah sakit, memindahkan barang saat bongkar muat, dan pekerjaan-pekerjaan lain yang sejenis.

## 3. Mobile Manipulator

Tipe mobile manipulator mengkombinasikan mobilitas dan manipulasi. Robot humanoid menirukan bagian-bagian tubuh dari manusia. Mobile manipulator dapat menggunakan effector-nya jauh lebih baik daripada yang dapat dilakukan anchored manipulator. Namun, pekerjaan mobile manipulator dibuat lebih sulit karena tidak memiliki kekakuan seperti yang dimiliki oleh anchor.

### **C. Robot Hardware**

Kesuksesan pembuatan sebuah robot bergantung pada desain sensor dan effector sehingga cocok untuk berbagai jenis pekerjaan.

#### 1. Sensor

Sensor adalah perceptual interface antara robot dengan lingkungannya. Sensor dibagi menjadi 2 kategori yaitu :

- a. Passive sensor yaitu pengamat yang sesungguhnya dari lingkungan. Mereka mengambil sinyal yang di-generate oleh source lain pada lingkungan. Contoh : kamera.

- b. Active sensor yaitu mengirimkan energi kepada lingkungan, mereka bergantung pada fakta bahwa energi ini dipantulkan kembali menuju ke sensor. Contoh : sonar.

Active sensor cenderung memberikan informasi yang lebih banyak daripada passive sensor, akan tetapi ketika multiple active sensor digunakan pada satu waktu, maka dapat terjadi kenaikan konsumsi power dan bahaya dari berbagai gangguan.

Selain active sensor dan passive sensor, dikenal pula istilah range finder yaitu sensor untuk memperkirakan jarak dengan obyek terdekat, misalnya PING dan SRF04.

## 2. Effector

Effector adalah sarana robot untuk bergerak dan mengubah bentuk body. Untuk memahami perancangan effector lebih lanjut, perlu dipahami motion dan bentuk secara abstrak menggunakan degree of freedom (DOF). Dalam penghitungan nDOF untuk setiap arah independen dari salah satu effector-nya yang dapat bergerak.

## D. Arsitektur Software Robot

Software architecture adalah sebuah metodologi untuk algoritma yang terstruktur. Sebuah arsitektur memuat bahasa dan tools yang digunakan untuk menuliskan program. Sama seperti filosofi mengenai mekanisme program dapat dibangun bersama-sama. Software architecture untuk robot saat ini harus mampu memutuskan bagaimana mengkombinasikan motive control dan deliberative planning berbasiskan model.

Kebanyakan arsitektur robot menggunakan teknik reactive pada level control yang lebih rendah dan teknik deliberative yang lebih tinggi. Arsitektur yang mengkombinasikan antara teknik reactive dan deliberative disebut hybrid architecture. Selain hybrid architecture yang merupakan arsitektur gabungan tersebut, ada 3 jenis arsitektur yang sering digunakan yaitu :

1. Subsumption Architecture yaitu sebuah framework untuk membuat kontroler reactive di luar dari finite state machine. Node yang ada pada mesin ini dapat memuat pengujian untuk variabel sensor tertentu. Dengan berbagai macam alasan yang ada, arsitektur ini merupakan arsitektur yang paling jarang digunakan di robotika, meskipun secara histori tingkat kepentingannya cukup besar. Namun, arsitektur ini merupakan pengaruh yang baik untuk arsitektur dan komponen individual dari beberapa arsitektur.

2. Three Layer Architecture

Hybrid architecture mengkombinasikan reaction dan deliberation. Hybrid architecture yang paling populer adalah three layer architecture. Arsitektur ini terdiri dari : reactive layer, executive layer, dan deliberative layer.

- Reactive layer, menyediakan kontrol law level untuk robot. Reactive layer memiliki karakteristik khusus yaitu adanya tight sensor. Putaran keputusan seringnya diukur menggunakan milisecond.
- Executive layer, sering disebut sequencing layer yang berperan sebagai perekat antara reactive layer dengan deliberative layer. Layer ini menerima instruksi dari deliberative layer dan melanjutkannya ke reactive layer.
- Deliberative layer, layer ini men-generate penyelesaian global untuk task yang kompleks menggunakan planning. Karena men-generate

penyelesaian termasuk di dalam kompleksitas komputasional, putaran pengambil keputusan diukur dalam hitungan menit. Deliberative layer menggunakan model untuk pengambilan keputusan. Model ini bisa merupakan hasil dari pembelajaran terhadap data atau bisa memanfaatkan informasi kondisi yang digabungkan di dalam executive layer.

### 3. Pipeline Architecture

Pipeline architecture mirip dengan subsumption, yang mana arsitektur ini mengeksekusi multiple process secara paralel. Namun demikian, modul yang spesifik dalam arsitektur ini menyerupai yang ada juga pada Three Layer Architecture. Hal yang ditekankan pada arsitektur ini adalah bahwa semua terjadi secara paralel. Ketika perception layer memmekanisme sensor data yang paling baru, control layer mendasarkan pilihannya pada data yang lebih lama. Dalam hal ini yang seperti ini, pipeline architecture sedikit mirip dengan otak manusia.

Secara umum, untuk mengemudikan robot ada beberapa teknik, yaitu :

1. Single wheel drive, dimana hanya satu roda depan yang dapat bergerak ke kanan dan ke kiri sebagai kemudi
2. Differential drive, dimana 2 roda di belakang dapat mengatur arah gerak robot.
3. Synchro drive, dimana 3 roda dapat mengemudikan robot.
4. Ackerman steering, dimana gerak robot dikendalikan oleh 2 roda depan dan 2 roda belakang.
5. Omni directional drive, dimana gerak robot dapat dikendalikan dari 3 atau 4 sistem roda yang dapat berputar ke segala arah, sehingga orientasi robot

tetap. Roda yang digunakan sering disebut sebagai omni wheel (3 roda) sehingga meskipun bergerak menghindari, namun orientasi gerak robot tetap dengan sudut roda standar.

## **E. Aplikasi Robot**

Ada beberapa domain aplikasi yang utama untuk teknologi robotika antara lain :

### 1. Industri dan pertanian

Robot yang ada pada bidang ini sebenarnya ditujukan untuk menggantikan pekerjaan kasar. Pekerjaan yang dimaksud harus terstruktur sehingga dapat diterima pada otomasi robot. Contoh : penempatan alat, pemindahan material, pengecatan dan lainnya.

### 2. Transportasi

Banyak sekali contoh robot yang ditempatkan untuk membantu manusia di bidang transportasi. Contoh : Helikopter otomatis untuk mengantarkan muatan ke tempat yang sulit dijangkau manusia, kursi roda otomatis untuk orang yang kesulitan menggerakkan roda.

### 3. Mobil robotik

Telah dikembangkan robot yaitu Boss dan Stanley tidak hanya untuk membuat pengemudi lebih aman, namun mobil robotik ini juga mampu membebaskan pengemudi melakukan pekerjaan di dalam mobil.

### 4. Perawatan kesehatan

Kini, semakin banyak robot yang digunakan untuk membantu dokter bedah dalam penempatan alat ketika operasi organ seperti otak, mata dan hati. Selain itu, di luar area kedokteran, sudah mulai dikembangkan robot untuk membantu orang cacat.

5. Lingkungan yang berbahaya

Robot dapat membantu manusia membersihkan sampah kimia nuklir.

6. Eksplorasi

Contoh yang paling mudah mengenai eksplorasi adalah yang paling sering dilakukan oleh NASA dengan mengirim robot untuk misi luar angkasa.

7. Pelayan Pribadi

Service robot dapat membantu individu melakukan pekerjaan sehari-hari.

8. Entertainment

Robot juga mulai digunakan dalam bidang entertainment. Contohnya : robot soccer, sebuah permainan sejenis sepakbola yang mempertandingkan antar robot dengan aturan permainan yang mirip dengan sepakbola.

9. Human Augmentation

Domain aplikasi akhir pada teknologi robotika adalah human augmentation. Robotic teleoperation atau telepresence adalah bentuk lain dari human augmentation. Teleoperation mampu membawa tugas dalam jarak jauh dengan bantuan dari device robot.

# BAB 14

## Jaringan Saraf Tiruan (*Artificial Neural Network*)

Setelah mempelajari Bab ini, Anda diharapkan mempunyai pengetahuan untuk :

- ❖ Menjelaskan tentang sejarah jaringan saraf tiruan
- ❖ Menjelaskan tentang definisi jaringan saraf tiruan
- ❖ Menjelaskan tentang konsep pemodelan jaringan saraf tiruan
- ❖ Menjelaskan tentang kelebihan dan kelemahan jaringan saraf tiruan
- ❖ Menjelaskan tentang arsitektur jaringan saraf tiruan
- ❖ Menjelaskan tentang fungsi aktivasi dan fungsi pembelajaran jaringan saraf tiruan

### A. Sejarah Jaringan Saraf Tiruan

Perkembangan jaringan saraf tiruan (*artificial neural network*) sudah ada sejak tahun 1943 ketika Warren McCulloch dan Walter Pitts memperkenalkan perhitungan model neural network. Mereka melakukan kombinasi beberapa processing unit sederhana bersama-sama yang mampu memberikan peningkatan secara keseluruhan pada kekuatan komputasi.

Kemudian dilanjutkan pada penelitian yang dilakukan oleh Rosenblatt pada tahun 1950 dimana ia berhasil menemukan sebuah two layer network, yang disebut sebagai perceptron. Perceptron memungkinkan untuk pekerjaan klasifikasi pembelajaran tertentu dengan penambahan bobot pada setiap koneksi antar network.

Keberhasilan perceptron dalam pengklasifikasian pola tertentu ini tidak sepenuhnya sempurna karena masih ditemukan juga beberapa keterbatasan. Perceptron tidak mampu menyelesaikan menyelesaikan problem XOR (exclusive or). Penilaian terhadap keterbatasan neural network ini membuat penelitian di bidang ini sempat mati selama kurang lebih 25 tahun. Namun perceptron berhasil menjadi sebuah dasar untuk penelitian selanjutnya di bidang jaringan saraf tiruan.

Pengkajian pada kecerdasan buatan mulai berkembang lagi di awal tahun 2980-an. Para peneliti banyak menemukan bidang tema menarik tentang ilmu jaringan saraf tiruan. Untuk saat ini jaringan saraf tiruan dapat diterapkan pada beberapa tugas antara lain : klasifikasi, pengenalan pola, prediksi, analisis klaster, simulasi dan banyak tugas ain yang berbeda.

## **B. Definisi Jaringan Saraf Tiruan**

Jaringan saraf tiruan yaitu suatu pengolahan informasi dengan meniru sistem saraf biologis seperti mekanisme pengambilan keputusan pada otak manusia. Jaringan syaraf tiruan adalah prosedur komputasi yang meniru sistem jaringan saraf biologis pada manusia.

Jaringan saraf tiruan merupakan konsep rekayasa pengetahuan dalam bidang kecerdasan buatan yang didesain dengan mengadopsi sistem saraf manusia, dimana pemrosesan utama sistem saraf manusia ada di otak. Bagian terkecil dari otak manusia adalah sel saraf yang menjadi unit dasar dalam memmekanisme informasi. Unit pemroses informasi ini dinamakan neuron, dimana terdapat 10 miliar neuron dalam otak manusia dan terdapat 60 triliun koneksi (sinapsis) antar neuron dalam otak manusia. Dengan neuron tersebut secara bersamaan, maka otak manusia dapat

memmekanisme informasi secara paralel dan cepat, bahkan lebih cepat dari komputer tercepat saat ini.

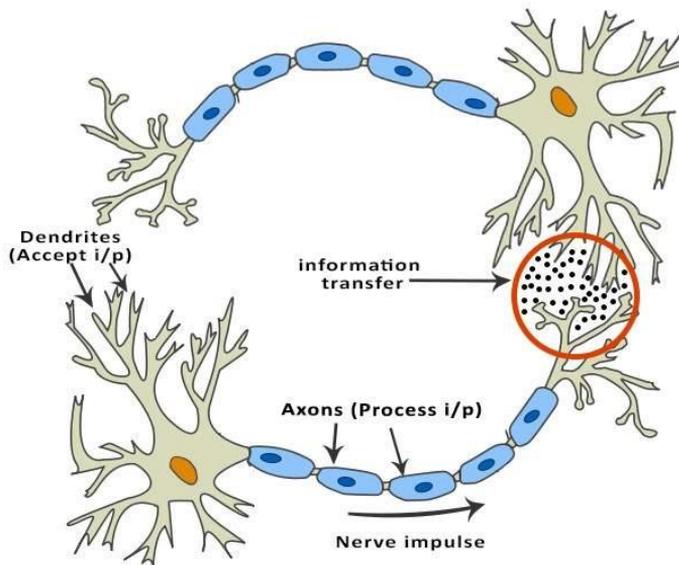
Layaknya, neuron biologi, jaringan saraf tiruan juga merupakan sistem fault tolerant dalam 2 hal, pertama dapat mengenali sinyal input yang agak berbeda dari yang pernah diterima sebelumnya. Sebagai contoh, manusia sering dapat mengenali seseorang yang wajahnya pernah dilihat dari foto atau tidak menjumpainya. Kedua, tetap mampu bekerja meskipun beberapa neuron tidak mampu bekerja dengan baik. Apabila sebuah neuron rusak, maka neuron lain dapat dilatih untuk menggantikan fungsi neuron yang rusak tersebut.

Jaringan saraf tiruan seperti manusia, belajar dari contoh karena mempunyai karakteristik yang adaptif, yaitu dapat belajar dari data-data sebelumnya dan mengenal pola data yang selalu berubah. Selain itu, jaringan saraf tiruan merupakan sistem yang tidak terprogram, artinya semua keluaran atau kesimpulan yang ditarik oleh jaringan didasarkan pada pengalamannya selama mengikuti mekanisme pembelajaran / pelatihan.

Secara biologis untuk sistem saraf pada manusia, terdiri dari elemen-elemen badan sel (soma), sejumlah serat yang menyalurkan informasi ke neuron (dendrit), dan sebuah serat tunggal yang keluar dari neuron (akson). Setiap sinyal luar yang diterima oleh dendrit akan melewati sinapsis untuk diteruskan ke neuron kemudian dimekanisme di dalam soma, setelah selesai akan dikeluarkan melalui akson untuk dimekanisme kembali oleh neuron yang lain ataupun keluar sebagai sinyal akhir hasil mekanisme di otak.

Dengan analogi seperti sistem kerja saraf otak pada manusia jaringan saraf tiruan terdiri dari unit pemroses yang disebut neuron berisi addder dan fungsi aktivasi, sejumlah bobot (sinapsis dalam otak manusia), sejumlah vektor masukan

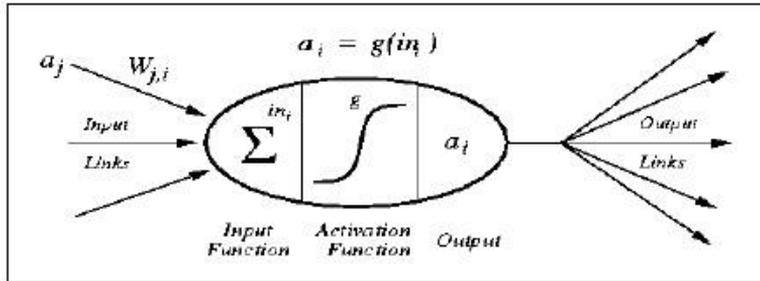
(dendrit dalam otak manusia). Fungsi aktivasi berguna untuk mengatur keluaran yang diberikan oleh neuron.



Cara kerja jaringan saraf tiruan seperti cara kerja manusia yaitu belajar melalui contoh. Suatu aplikasi jaringan saraf tiruan dikonfigurasi untuk aplikasi tertentu, seperti pengenalan pola atau klasifikasi data, melalui mekanisme pembelajaran.

### C. Konsep Pemodelan Jaringan Saraf Tiruan

Jaringan saraf pada manusia ditiru pola pemrosesannya yang dimanifestasikan sebagai sebuah neuron. Sinyal masukan dikalikan dengan masing-masing penimbang yang bersesuaian. Kemudian dilakukan penjumlahan dari seluruh hasil perkalian sehingga keluaran yang dihasilkan



#### D. Kelebihan Jaringan Saraf Tiruan

Kelebihan jaringan saraf tiruan antara lain :

1. Belajar adaptif, kemampuan dalam mempelajari bagaimana melakukan pekerjaan berdasarkan data yang diberikan untuk pelatihan atau pengalaman awal.
2. *Self organization*, sebuah jaringan saraf tiruan dapat membuat organisasi sendiri atau representasi dari informasi yang diterimanya selama pembelajaran (prosedur *training*).
3. *Real Time Operation* : Perhitungan JST dapat dilakukan secara paralel sehingga perangkat keras yang dirancang dan diproduksi secara khusus dapat mengambil keuntungan dari kemampuan ini.

#### E. Kelemahan Jaringan Saraf Tiruan

Kelemahan jaringan saraf tiruan antara lain :

1. Tidak efektif apabila digunakan untuk melakukan operasi-operasi numerik dengan presisi tinggi.
2. Tidak efisien apabila digunakan untuk melakukan operasi algoritma aritmatika, operasi logika dan simbolis.

3. Untuk beroperasi, jaringan saraf tiruan memerlukan pelatihan / pembelajaran (*training*) sehingga apabila jumlah datanya besar, waktu yang digunakan untuk mekanisme pelatihan sangat lama.

## **F. Arsitektur Jaringan Saraf Tiruan**

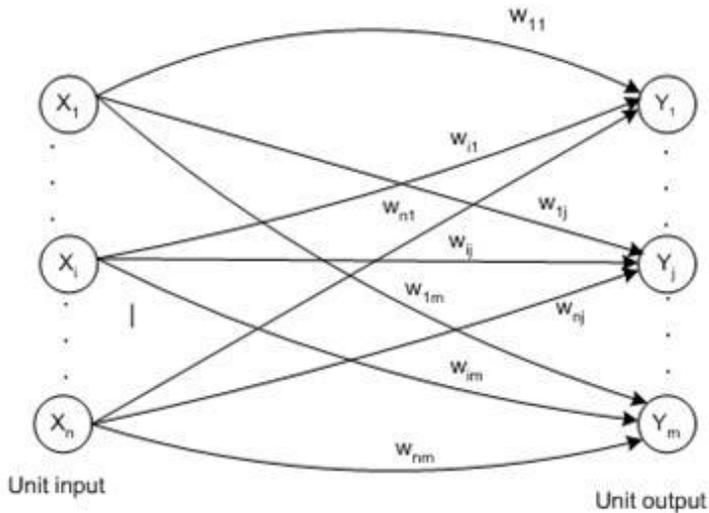
Baik tidaknya suatu model jaringan saraf tiruan ditentukan oleh hubungan antar neuron atau arsitektur jaringan saraf tiruan itu sendiri. Neuron terkumpul dalam lapisan yang disebut neuron layer yang terbagi menjadi tiga lapisan :

1. Lapisan Input (*Input Layer*), unit dalam lapisan input ini disebut unit input yang bertugas menerima pola inputan dari luar yang menggambarkan suatu permasalahan
2. Lapisan tersembunyi (*hidden layer*), unit dalam lapisan tersembunyi disebut unit tersembunyi, yang mana nilai output tidak dapat diamati secara langsung.
3. Lapisan output (*Output Layer*), unit dalam lapisan output disebut unit output yang merupakan penyelesaian jaringan saraf tiruan terhadap suatu permasalahan.

Beberapa arsitektur jaringan saraf tiruan yang sering dipakai meliputi :

1. Jaringan Lapisan Tunggal (*Single Layer Network*)

Dalam jaringan ini, sekumpulan input neuron dihubungkan langsung dengan sekumpulan outputnya. Dalam beberapa model (misalnya perseptron), hanya ada sebuah unit neuron output:



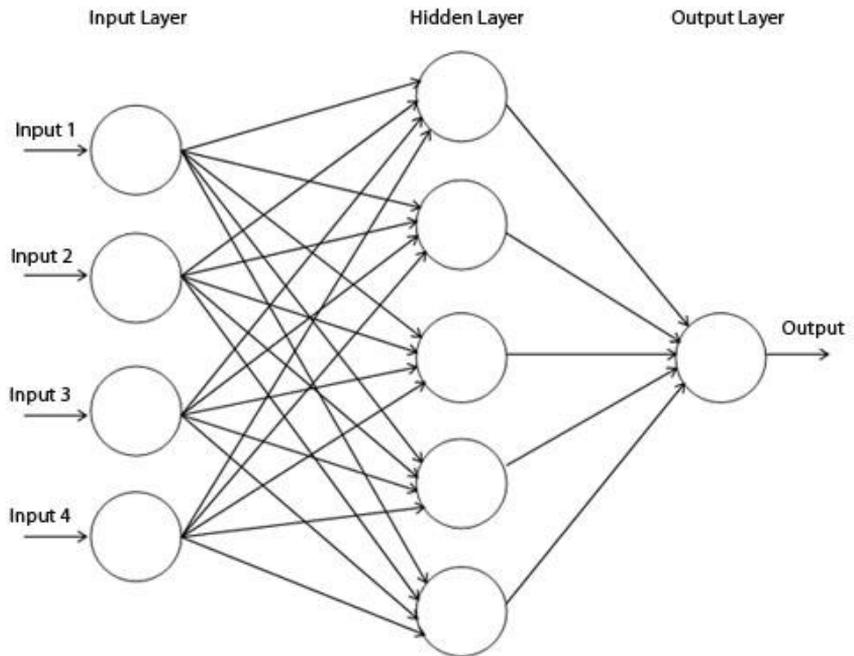
Dalam jaringan ini, semua unit input dihubungkan dengan semua unit output, meskipun dengan bobot yang berbeda-beda. Tidak ada unit input yang dihubungkan dengan unit input lainnya. Demikian pula dengan unit outputnya.

Besaran  $w_{ji}$  menyatakan bobot hubungan antara unit ke- $i$  dalam input dengan unit ke- $j$  dalam output. Bobot-bobot ini saling independen. Selama mekanisme pelatihan, bobot-bobot tersebut akan dimodifikasi untuk meningkatkan keakuratan hasil. Model semacam ini tepat digunakan untuk pengenalan pola karena kesederhanaannya.

## 2. Jaringan Layar Jamak (*Multi Layer Network*)

Jaringan layar jamak merupakan perluasan dari layar tunggal. Dalam jaringan ini, selain unit input dan output, ada unit-unit lain yang disebut hidden layer atau lapisan tersembunyi. Dimungkinkan pula ada beberapa

layer tersembunyi, sama seperti pada unit input dan output, unit-unit dalam satu layer tidak saling berhubungan.



Gambar diatas adalah jaringan dengan  $n$  buah unit input ( $X_1, X_2, \dots, X_n$ ), sebuah layer tersembunyi yang terdiri dari  $p$  buah unit ( $Z_1, Z_2, \dots, Z_n$ ) dan  $m$  buah unit output ( $Y_1, Y_2, \dots, Y_m$ ).

Jaringan layer jamak dapat menyelesaikan problem yang lebih kompleks dibandingkan dengan layer tunggal, meskipun kadangkala mekanisme pelatihan lebih kompleks dan lama.

## G. Fungsi Aktivasi Jaringan Saraf Tiruan

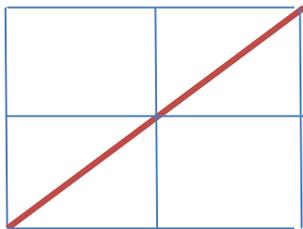
Dalam jaringan saraf tiruan, fungsi aktivasi dipakai untuk menentukan keluaran suatu neuron. Argumen fungsi aktivasi adalah net masukan (kombinasi linier masukan dan bobotnya). Apabila  $net = \sum x_i \cdot w_i$  maka fungsi aktivasinya adalah  $f(net) = f(\sum x_i \cdot w_i)$ .

Beberapa fungsi aktivasi yang sering dipakai adalah sebagai berikut :

1. Fungsi aktivasi linear

Fungsi aktivasi ini biasanya digunakan untuk keluaran jaringan saraf tiruan yang nilai keluarannya diskret. Apabila  $v$  adalah nilai gabungan dari semua vektor oleh adder, maka sinyal keluaran  $y$  didapatkan dengan memberikan nilai  $v$  apa adanya untuk mejadi nilai keluaran. Tidak ada perhitungan apapun yang dilakukan pada nilai  $v$  yang didapat sehingga umumnya jaringan saraf tiruan yang menggunakan fungsi aktivasi ini bisa dianggap tidak menggunakan fungsi aktivasi. Nilai  $y$  sebagai nilai keluaran, diformulasikan seperti pada persamaan berikut :

$$y = \text{sign}(v) = v$$



2. Fungsi aktivasi step

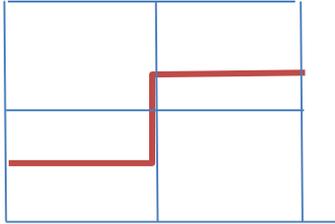
Apabila  $v$  adalah nilai gabungan dari semua vektor oleh adder, maka keluaran  $y$  didapatkan dengan memberikan ambang batas pada nilai  $v$  berdasarkan nilai  $T$  yang diberikan. Nilai  $y$  diformulasikan sebagai berikut :

$$y = \text{sign}(v) = \begin{cases} 1 & \text{jika } v \geq T \\ -1 & \text{jika } v < T \end{cases}$$

Bentuk diatas disebut juga ambang batas bipolar.

Sedangkan ambang batas biner diformulasikan sebagai berikut :

$$y = \text{sign}(v) = \begin{cases} 1 & \text{jika } v \geq T \\ 0 & \text{jika } v < T \end{cases}$$



### 3. Fungsi aktivasi sigmoid biner

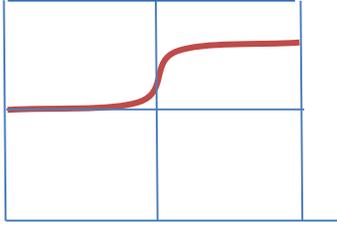
Untuk jaringan saraf tiruan dengan nilai keluaran kontinu biasanya menggunakan fungsi aktivasi sigmoid. Nilai sinyal keluaran  $y$  dihitung menggunakan fungsi kurva sigmoid dengan interval nilai keluaran mulai 0 sampai 1.

Nilai  $y$  diformulasikan sebagai berikut :

$$y = \text{sign}(v) = \frac{1}{1 + e^{-\alpha v}}$$

parameter  $\alpha$  adalah parameter kemiringan (slope) pada kurva sigmoid yang dihasilkan. Semakin besar nilai  $\alpha$ , maka semakin tegak kurva yang diberikan dan semakin kecil kurva maka semakin landai kurva yang diberikan. Umumnya nilai  $\alpha$  yang digunakan adalah 1 sehingga formula yang umum digunakan menjadi :

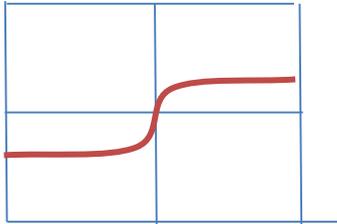
$$\frac{1}{1 + e^{-v}}$$



#### 4. Fungsi aktivasi sigmoid bipolar

Fungsi aktivasi ini sebenarnya sama dengan sigmoid biner, hanya saja batas nilai keluaran yang diberikan adalah -1 sampai 1. Fungsi aktivasi ini sangat baik digunakan untuk mekanisme klasifikasi karena lebih tangguh dalam menangani data-data yang banyak didominasi oleh nilai nol. Nilai  $y$  diformulasikan sebagai berikut :

$$y = \text{sign}(v) = \frac{1}{1+e^{-av}} - 1$$



Nilai parameter  $\alpha$  yang digunakan umumnya 1 sehingga formula diatas berubah menjadi seperti berikut :

$$y = \frac{2}{1+e^{-v}} - 1 - 1$$

## H. Fungsi Pembelajaran

Jaringan saraf tiruan akan mencoba melakukan simulasi kemampuan otak manusia untuk belajar. Jaringan saraf tiruan tersusun atas neuron-neuron dan dendrit. Tidak seperti model biologis, jaringan saraf memiliki struktur yang tidak dapat diubah, dibangun oleh sejumlah neuron, dan memiliki nilai tertentu yang menunjukkan seberapa besar koneksi antara neuron (bobot). Perubahan yang terjadi selama mekanisme pembelajaran adalah perubahan nilai bobot. Nilai bobot akan bertambah, apabila informasi yang diberikan oleh neuron tersampaikan, sebaliknya apabila informasi tidak disampaikan oleh suatu neuron ke neuron yang lain, maka nilai bobot akan diubah secara dinamis hingga mencapai suatu nilai yang cukup seimbang. Apabila nilai ini telah tercapai mengindikasikan bahwa tiap-tiap input telah berhubungan dengan output yang diharapkan.

Ada dua jenis pembelajaran pada jaringan saraf tiruan meliputi :

### 1. Jaringan terawasi (*supervised learning*)

Prosedur pembelajaran pada jaringan saraf disebut terawasi apabila output yang diharapkan telah diketahui sebelumnya.

Contoh : andaikan kita memiliki jaringan saraf yang akan digunakan untuk mengenali pasangan pola, misalkan pada operasi “AND” berikut :

Input		Target
0	0	0
0	1	0
1	0	0
1	1	0

Pada mekanisme pembelajaran, satu pola akan diberikan ke satu neuron pada lapisan input. Pola ini akan dirambatkan di sepanjang jaringan saraf hingga sampai ke neuron pada lapisan output. Lapisan output ini akan membangkitkan pola output yang nantinya akan dicocokkan dengan pola output targetnya.

Apabila terjadi perbedaan antara pola output hasil pembelajaran dengan pola target, maka akan muncul error. Apabila nilai error ini masih cukup besar, mengindikasikan bahwa masih perlu dilakukan lebih banyak pembelajaran lagi. Beberapa jenis pembelajaran terawasi dapat menggunakan beberapa cara atau bentuk antara lain :

- a. Hebb Rule
- b. Perceptron
- c. Delta Rule
- d. Backpropagation
- e. Heteroassociative memory
- f. Bidirectional Associative Memory (BAM)
- g. Learning Vector Quantization (LVQ)

## 2. Jaringan tak terawasi (*unsupervised learning*)

Pada model pembelajaran tak terawasi ini tidak memerlukan target output. Pada prosedur ini, tidak dapat ditentukan hasil yang seperti apakah yang diharapkan selama mekanisme pembelajaran. Selama mekanisme pembelajaran, nilai bobot disusun dalam suatu range tertentu tergantung pada nilai input yang diberikan. Tujuan pembelajaran ini adalah mengelompokkan unit-unit yang hampir sama dalam suatu area tertentu. Pembelajaran ini biasanya sangat cocok untuk pengelompokan (klasifikasi) pola.

Jaringan ini pertama kali diperkenalkan oleh Prof. Teuvo Kohonen pada tahun 1982. Pada jaringan ini, suatu lapisan yang berisi neuron-neuron akan menyusun dirinya sendiri berdasarkan input nilai tertentu dalam suatu kelompok yang dikenal dengan istilah cluster. Selama mekanisme penyusunan diri, cluster yang memiliki vektor bobot paling cocok engan pola input (memiliki jarak yang paling dekat), akan terpilih sebagai pemenang. Neuron yang menjadi pemenang beserta neuron-neuron tetangganya akan memperbaiki bobot-bobotnya.

## **DAFTAR PUSTAKA**

- Aries Syamsuddin, 2004. Pengenalan Algoritma Genetik. Ilmukomputer.com
- Eng Agus Naba, Belajar Cepat Fuzzy Logic Menggunakan Matlab, Yogyakarta : Andi Offset
- Jong Jek Seng, Jaringan Syaraf Tiruan dan Pemrogramannya Menggunakan Matlab, Yogyakarta : Andi Offset
- Muhammad Dahria, Kecerdasan Buatan (Artificial Intelligence), Jurnal Saintikom Vol. 5 No. 2 Agustus 2008.
- Sri Kusumadewi, Artificial Intelligence, Jakarta : Penerbit Graha Ilmu
- T. Sutojo, Mulyanto, V. Suhartono, Kecerdasan Buatan, Yogyakarta : Andi Offset
- Turban et.al, Decision Support System and Intelligence System, Jilid 2, Yogyakarta : Andi Offset
- Widodo Budiharto dan Derwin Suhartono, Artificial Intelligence Konsep dan Pengaplikasiannya, Yogyakarta : Andi Offset



**YAYASAN PRIMA AGUS TEKNIK**

ISBN 978-623-6141-31-1 (PDF)



9 786236 141311