

8. Kaedah Kepintaran Buatan

8.0 AI sebagai strategi gelintaran

8.1 Manipulasi simbolan

8.2 Deduksi dan keputusan

8.3 Induksi dan pembelajaran

8.4 Penjanaan

- Tugas 14
- Tugas 15

8.0 AI sebagai strategi gelintaran

8.0.0 Kepintaran Buatan (AI)

Kepintaran buatan ialah usaha untuk memesinkan telatah kepintaran. Isu dalam kepintaran buatan itu berkait dengan pemahaman kepintaran itu sendiri, dan kaedah untuk memesinkannya,

$$\text{Kepintaran Buatan} = \text{Kepintaran} + \text{Buatan}$$

8.0.1 Kepintaran

Mula-mula perlu dijelaskan apa itu kepintaran.

Sekurang-kurangnya, telatah pintar dikenali. Di antaranya:

- belajar atau memahami dari pengalaman
- memahami perutusan-perutusan kabur dan saling bertentangan
- Bertindakbalas dengan cepat dan berjaya dalam situasi baru (gerakbalas penyesuaian, berbeza)
- Gunakan pemikiran dalam penyelesaian masalah
- memperoleh dan menggunakan pengetahuan
- dan lain-lain

Bahkan, didepankan suatu ujian untuk kepintaran yang berdasarkan telatah pintar, iaitu Ujian Turing:

Ujian Turing:

Sesuatu komputer itu pintar jika apabila seseorang berkomunikasi dengan komputer itu tanpa melihatnya, dan dengan manusia tanpa melihatnya, dia tidak dapat menentukan yang mana yang mana.

Untuk memberikan telatah kepintaran, kajian kepintaran buatan tradisi menentukan yang berikut sebagai ciri-ciri sistem AI:

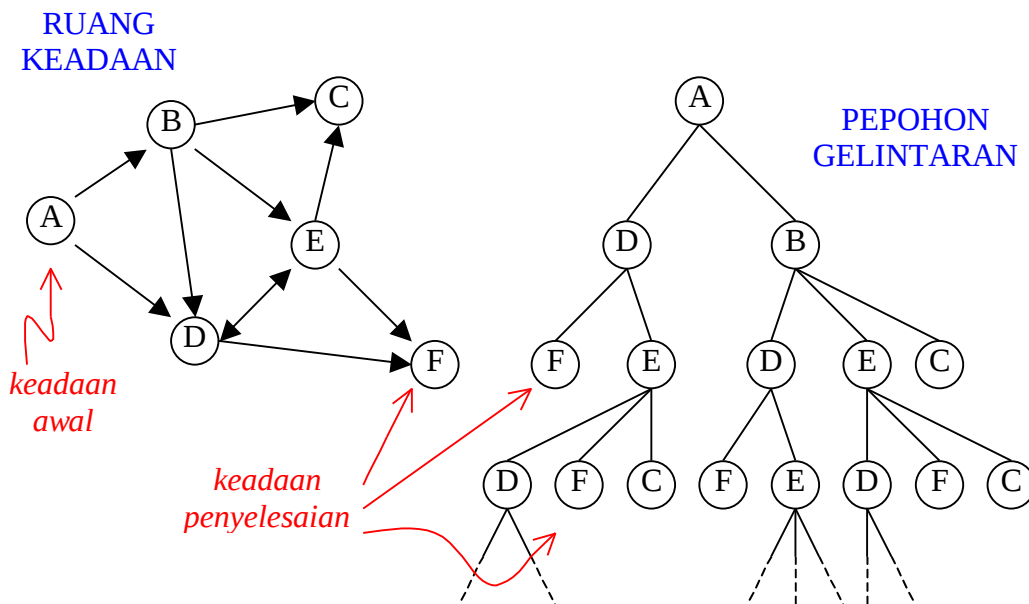
- pemprosesan simbolan (simbol mewakili konsep dalam masalah) lebih daripada pemprosesan angka
- heuristik ('peraturan ibujari') lebih daripada algoritma
- kesimpulan dari fakta dan peraturan
- pepadanan corak (mengenai objek, peristiwa, proses, terhadap kepada ciri kualitatif dan hubungan logik, dan pengiraannya)

8.0.2 Penyelesaian Masalah

Penyelesaian masalah, suatu proses pintar, membabitkan ruang masalah yang berkaitan. Ruang masalah – juga dirujuk sebagai ruang keadaan – mengandungi keadaan yang dibenarkan. Keadaan-keadaan ini diwakili oleh nod, dan gerakan yang dibenarkan di antara keadaan-keadaan (perubahan keadaan yang dibolehkan dalam perihalan masalah) diwakili oleh pautan yang berarah di antara nod-nod berkenaan.

Penyelesaian masalah boleh dilihat sebagai gelintaran dalam ruang masalah. Bermula di nod yang mewakili keadaan awal, ruang keadaan digelintar untuk mencapai nod keadaan akhir atau penyelesaian yang dikehendaki, menerusi gerakan yang dibenarkan.

Proses gelintaran atau carian boleh dirajahkan sebagai pepohon (disebut pepohon gelintaran), yang nodnya merupakan keadaan-keadaan, dan nod daun dari nod tertentu adalah keadaan-keadaan yang dapat dicapai secara langsung melalui gerakan yang dibenarkan dari keadaan yang diwakili oleh nod ibu. Contoh ruang keadaan dan pepohon gelintarannya ditunjukkan di bawah.



Nod akar pepohon gelintaran adalah keadaan awal dalam masalah; nod yang mewakili keadaan penyelesaian atau keadaan akhir adalah nod yang dicari.

Untuk masalah biasa, ruang keadaan terlalu besar. Kita tidak perlu rajahan keseluruhan ruang keadaan untuk mengungkap semua pergerakan. Kita boleh tulis peraturan umum untuk mewakili pergerakan yang dibenarkan ini, dan cabang pepohon gelintaran dijana hanya apabila diperlukan, menggunakan peraturan ini.

Pepohon gelintaran untuk masalah biasa juga biasanya terlalu besar. Dengan itu kita memerlukan strategi atau cara untuk melakukan carian yang tertentu supaya penyelesaiannya diperoleh dengan cepat dan cekap. Kaedah atau strategi gelintaran yang digunakan mencerminkan kepintaran sistem. Berikut dijadualkan kaedah-kaedah gelintaran.

Kaedah Gelintaran

	gelintaran buta		heuristik (petunjuk, intuisi, penilaian, hukum ibujari, inspirasi ...)	pengoptimuman (analisis)
	penyenaraian penuh	gelintaran separa		
Proses gelintaran	semak	semak	hanya	jana keadaan

	semua keadaan yang mungkin	beberapa kemungkinan, buang yang tak bagus	pergerakan dgn potensi baik dipertimbangkan	lebih baik secara analisis/ peroleh penyelesaian terbaik secara langsung
ujian	bandingan. Berhenti bila semua telah disemak	bandingan semasa	berhenti bila penyelesaian memuaskan	berhenti bila tiada pembaikan
penyelesaian	terbaik	terbaik di antara yg disemak	memuaskan	hampir terbaik/terbaik
wilayah AI				

Sesuatu kaedah gelintaran harus menyebabkan pergerakan (misalnya ia tidak tertangkap dalam kitar), dan bersistem (misalnya ia tidak pertimbang sesuatu keadaan berulang-ulang).

Sesuatu proses gelintaran itu boleh punyai satu dari dua arah gelintaran. Ia dikatakan dipacu data apabila punyai rantaian hadapan, di mana dimulakan dengan fakta (iaitu keadaan awal), dari itu diperolehi keadaan/fakta baru, menerusi gerakan dibenarkan, dan seterusnya, sehingga sampai penyelesaian (iaitu keadaan akhir). Ia dipacu sasaran apabila punyai rantaian kebelakang, di mana dimulakan dengan keadaan sasaran (iaitu penyelesaian yang dicari), kemudian diperolehi subsasaran menerusi gerakan ke belakang, dan seterusnya sehingga boleh ditunjukkan bahawa sasaran adalah hasil fakta atau keadaan-keadaan awal.

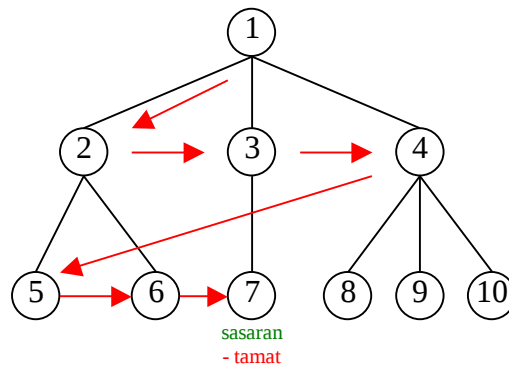
Gelintaran dipacu sasaran lebih cekap dalam kes sasaran mudah diformulasikan, misalnya dalam pembuktian teorem atau diagnosis perubatan, dan apabila peraturan terpakai (pergerakan dibenarkan) untuk fakta lebih banyak berbanding peraturan yang boleh menghasilkan sasaran, misalnya dalam pembuktian teorem. Ia juga berkelebihan apabila semua atau sebahagian data tidak diberikan, jadi ia kena diperolehi oleh penyelesai masalah bila diperlukan, misalnya dalam diagnosis perubatan. Keadaan sebaliknya untuk gelintaran dipacu data.

Gelintaran boleh dipandu ciri keadaan semasa yang diteliti, seperti yang menggunakan heuristik dan yang melakukan analisis. Gelintaran buta tidak mengambil peduli kualiti keadaan semasa.

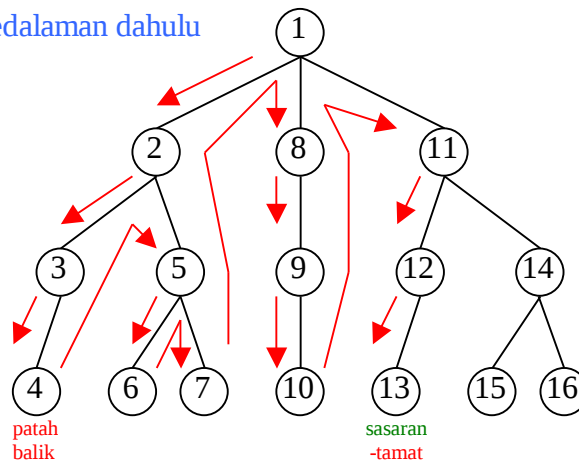
Gelintaran buta boleh jadi gelintaran penuh, di mana seluruh pepohon disemak, atau gelintaran separa, di mana ada dua arah asas menggelintar suatu pepohon – kelebaran dahulu, dan kedalaman dahulu. Gelintaran buta memerlukan pengiraan yang banyak, dan hanya sesuai untuk masalah kecil.

Kaedah-kaedah gelintaran kelebaran dahulu dan kedalaman dahulu diskemakan di bawah.

Gelintaran kelebaran dahulu



Gelintaran kedalaman dahulu



Ini pula algoritma berkenaan: gelintaran kelebaran dahulu,

```
Awalkan SENARAI_NOD
Sehingga nod sasaran ditemui atau SENARAI_NOD kosong, buat:
  Buang unsur pertama drp SENARAI_NOD dan namakannya E
  Bagi setiap peraturan terpakai utk E, buat:
    Kenakan peraturan utk dapat keadaan baru, F
    Jika F keadaan sasaran, keluar, dan kembalikan F
    Jika tidak, tambah F ke hujung SENARAI_NOD
```

dan, gelintaran kedalaman dahulu,

```
Jika keadaan awal keadaan sasaran, keluar dan kembalikan kejayaan
Jika tidak, lakukan yg berikut sehingga kejayaan atau kegagalan diperolehi::
  Jana E, keadaan keturunan kpd keadaan awal. Jika tiada (lagi), kembalikan kegagalan
  Panggil Gelintaran Kedalaman Dahulu dgn E sbg keadaan awal
  Jika kejayaan diperolehi, kembalikan kejayaan. Jika tidak, teruskan dalam gelung ini
```

Kelebihan gelintaran kelebaran dahulu ialah, ia tidak tertangkap dalam hujung mati. Ini berlebihan terutama jika pepohon gelintaran ada kitar dan/atau pepohon tak dalam. Kaedah ini menemui

penyelesaian jika ia wujud. Jika beberapa penyelesaian wujud, yang memerlukan bilangan tersedikit pergerakan ditemui (apabila kedalaman penyelesaian berbeza-beza). Kelebihan gelintaran kedalaman dahulu ialah kurang ingatan diperlukan – hanya nod teratas jejak semasa disimpan, dan ia menemui penyelesaian tanpa banyak gelintaran jika keadaan-keadaan keturunan diatur, dan beberapa penyelesaian wujud.

8.0.3 Perwakilan Pengetahuan

Keadaan-keadaan dan pergerakan di antaranya mencirikan sesuatu masalah. Inilah pengetahuan tentang masalah berkenaan.

Cara pengetahuan distor (dalam komputer) adalah penting. Ia perlu punyai sifat keterungkapan dan kecekapan. Dalam keterungkapan cara itu sebaiknya membolehkan perwakilan semua bentuk pengetahuan iaitu keadaan dengan bentuk-bentuk kelainan yang ada dan pergerakan antara keadaan, diberi kekangan dari senibina mesin dan perisian ke atas struktur yang dibolehkan. Dalam kecekapan pula, cara itu dapat menangani kekompleksan menerusi pemudahan: banyak keadaan, pergerakan, hubungan, dan lain-lain diwakili beberapa symbol yang minimal. Cara dipilih biasanya lebih mudah bila skema perwakilan punyai struktur tabii.

Ada pelbagai jenis pengetahuan, misalnya hubungan, warisan, taabiran, dan tatacara.

Diberikan beberapa skema perwakilan pengetahuan yang lazim.

8.0.3.1 Logik

Logik formal digunakan.

Contoh, dalam logik usulan, *hujan*, *tiada_payung*, *basah* mewakili keadaan-keadaan, dan $hujan \wedge tiada_payung \rightarrow basah$ mewakili gerakan. Begitu juga, dalam logik predikat, contoh gerakan ialah $burung(X) \leftarrow terbang(X), berparuh(X)$

8.0.3.2 Tatacara

Dalam kaedah ini, set arahan untuk selesaikan masalah mewakili gerakan.

Contoh dalam sistem berdasarkan hukum,

$IF\ hujan\ THEN\ guna_umbrella$

dan dalam sistem produksi,

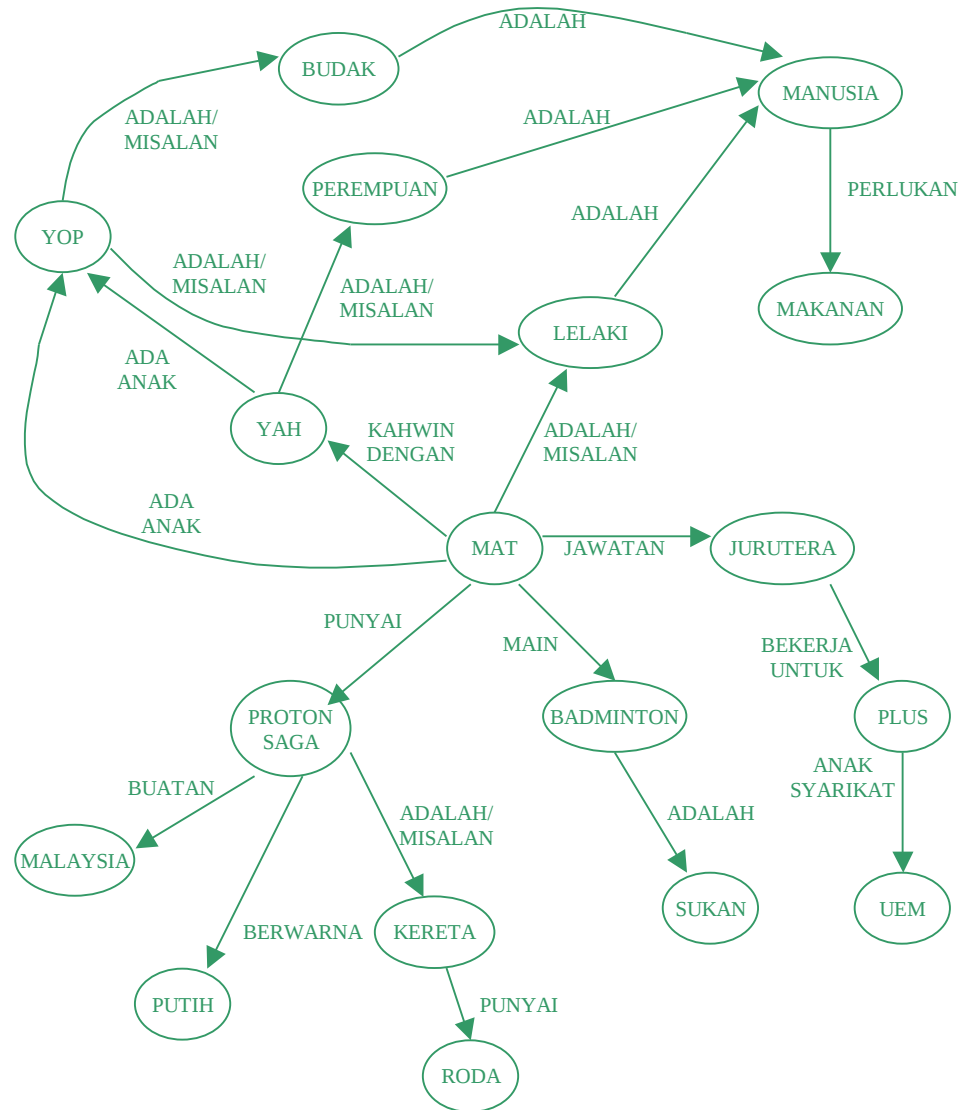
$buka_pintu \rightarrow pusing_tombol, tarik$

8.0.3.3 Rangkaian

Cara ini menggunakan graf.

Satu contoh ialah rangkaian atau jaring semantik, yang memberikan pengetahuan keadaan yang punyai komponen-komponen berhubungan.

Di bawah ialah suatu contoh rangkaian semantik.



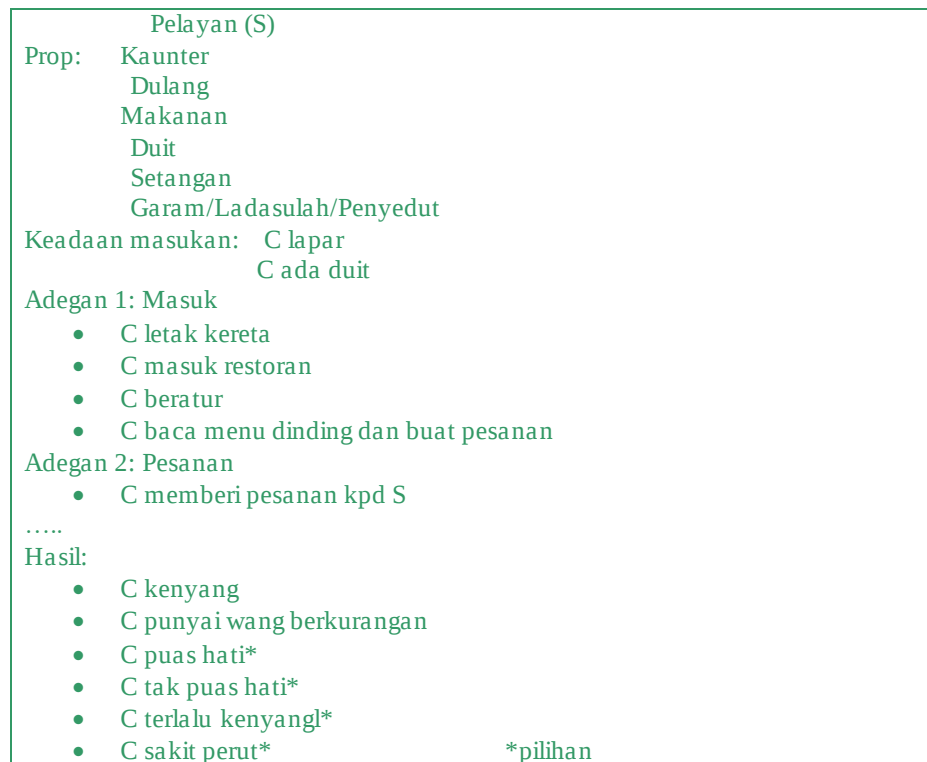
8.0.3.4 Berstruktur

Ini bagi struktur data kompleks. Contoh adalah skrip, bagi memerihalkan jujukan peristiwa yg stereotip, dan bingkai, yang menyerupai jaring semantik, tetapi dengan nod yang kompleks yang berbentuk slot dan pengisi.

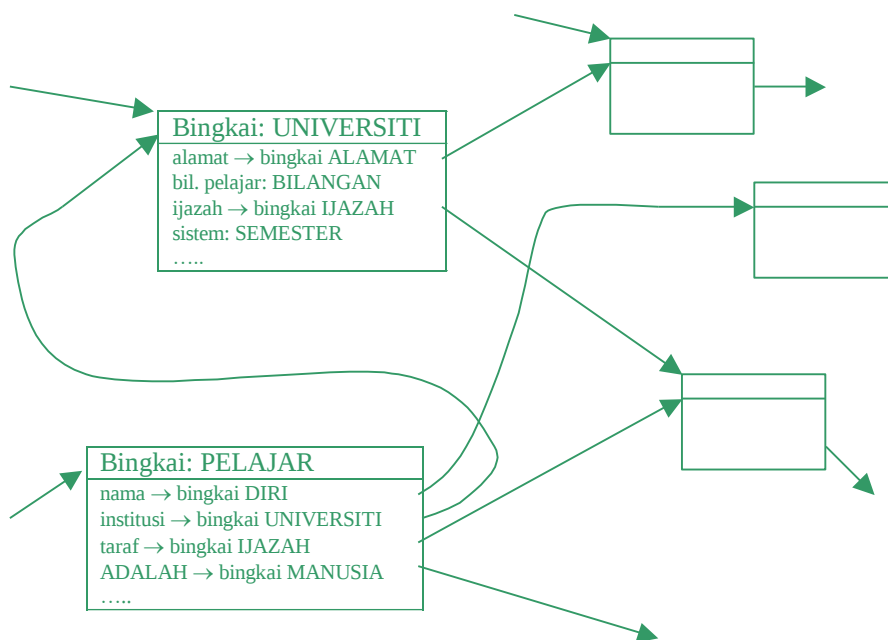
Berikut ialah suatu contoh skrip.

Skrip Restoran Makanan Segera

Pelakon: Pelanggan (C)



Ini pula suatu contoh bingkai.

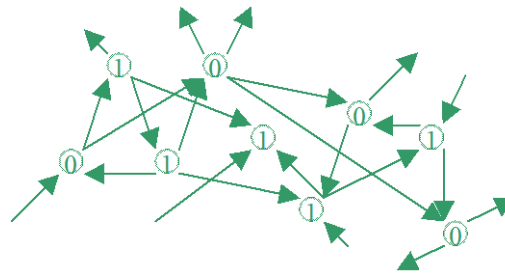


8.0.3.5 Subsimbolan

Di sini pengetahuan tidak diwakili oleh symbol. Ia menggunakan perwakilan yang lebih rendah.

Rangkaian neuronan misalnya, merupakan rangkaian unsur-unsur pengkomputeran mudah yang

berganding. Pengetahuan diwakilkan dalam pola unsur-unsur ini.



8.0.4 Kepintaran Buatan sebagai strategi gelintaran

Gelintaran lebih cekap boleh diperolehi menerusi strategi tertentu. Apa disebut kepintaran buatan boleh dikaitkan dengan kecekapan dan ketepatan yang lebih dalam penyelesaian masalah menerusi penggunaan strategi yang baik.

Strategi penggelintaran dijalankan menerusi penggunaan heuristik dan/atau analisis ruang masalah secara terus. Heuristik adalah cara penggelintaran yang cekap dan tepat daripada penggunaan ‘hukum ibujari’ (yang didapati membuat penggelintaran baik, tanpa hasil analisis jelas) tertentu. Ia juga biasa digunakan memanggil fungsi ‘kebaikan’ (misalnya, berapa dekat dengan penyelesaiannya sebenar) bagi keadaan yang dipertimbangkan.

Satu kaedah heuristik yang lazim adalah kaedah jana dan uji. Suatu keadaan akhir berpotensi (calon penyelesaian) dijana menurut gerakan yang sesuai. Ia diuji samada ia penyelesaian. Proses diulang jika tidak.

Satu heuristik dalam gelintaran ialah gelintaran terbaik dahulu, di mana dipilih nod terbaik (diukur oleh suatu fungsi heuristik) dahulu, samada menurut kelebaran atau kedalaman. Ada harapan penyelesaian terbaik diperolehi, dalam langkah yang sedikit, tetapi ini tidak terjamin.

Panjatan bukit ialah kaedah jana dan uji yang menurut kebaikan (diukur menurut fungsi heuristic diberi). Keadaan baru seterusnya yang terbaik dijana, atau dipilih daripada yang dijana. Panjatan bukit mudah ialah di mana keadaan yang dijana terus dipilih jika ia lebih baik daripada keadaan semasa. Dalam panjatan bukit tercerun pula dipertimbangkan semua keadaan yang boleh dihasilkan oleh keadaan semasa dan dipilih yang terbaik.

Analisis cara-matlamat ialah kaedah gelintaran yang merupakan campuran rantaian ke hadapan dan rantaian ke belakang. Lihat beza antara keadaan semasa dan keadaan sasaran semasa dan cuba kurangkan perbezaan ini. Ini memberikan submasalah daripada masalah asal.

Lihat contoh masalah jurujual jalanan. Dalam masalah ini, seorang jurujual perlu melawat N kota. Semua harus dilawati, setiapnya sekali. Diberi jarak-jarak di antara kota, dia perlu mengatur jujukan kota-kota dilawati, yang memberikan jumlah perjalanan yang paling pendek. Kekompleksan masalah ini ialah $(N-1)! \sim O(\exp N)$. Satu contoh heuristik ialah cabang-dan-batas: berhenti gelintar atas suatu keadaan apabila jejak jurujual yang sedang dibina jarak jumlah semasanya melebihi suatu had. Satu lagi heuristik yang lazim ialah heuristik jiran terdekat: dari suatu kota, pergi ke kota terdekat yang

belum dilawati (kekompleksan $O(N^2)$ sahaja). Yang kedua ini contoh algoritma tamak. Ia tidak dijamin memberikan penyelesaian terbaik, tetapi biasanya memberikan penyelesaian memuaskan.

Jika ruang masalah dapat diperihalkan secara analisis, pencarian keadaan terbaik boleh dibuat menggunakan kaedah analisis. Ini dipanggil juga proses pengoptimuman. Kebagusan keadaan diberikan secara analisis, dan penyelesaian diperolehi langsung menerusi analisis, atau menggunakan suatu algoritma yang meningkatkan kebagusan keadaan kepada nilai maksimum.

Satu contoh ialah rangkaian neuronan Hopfield, di mana neuron-neuron dedua V_i berganding,

$$V_i(\tau+1) = f\left(\sum_j T_{ij} V_j(\tau)\right) \equiv f(h_i(\tau)),$$

di mana τ ialah langkah masa, dan f adalah fungsi tangga. Fungsi Liapunov,

$$E = -\frac{1}{2} \sum_i \sum_j T_{ij} V_i V_j$$

menyusut secara ekanada terhadap dinamik neuron itu: ΔE samada negatif atau sifar. Bila h_i negatif, samada $V_i: 1 \rightarrow 0$, ΔE negatif, atau tak berubah pada nilai 0, ΔE sifar, dan bila h_i positif, samada $V_i: 0 \rightarrow 1$, ΔE negatif, atau tak berubah pada nilai 1, dan ΔE sifar. Nilai-nilai neuron mewakili keadaan-keadaan, dan gerakan yang diberikan persamaan dinamikanya mewakili pergerakan-pergerakan terbaik (menurut heuristik panjatan bukit) yang mencari penyelesaian yang diberikan oleh keadaan dengan nilai E terendah. Bahkan, secara analisis langsung, keadaan penyelesaian ialah yang punyai E minimum, dan bolehlah diselesaikan bagi V_i untuk nilai-nilai T_{ij} dipakai.

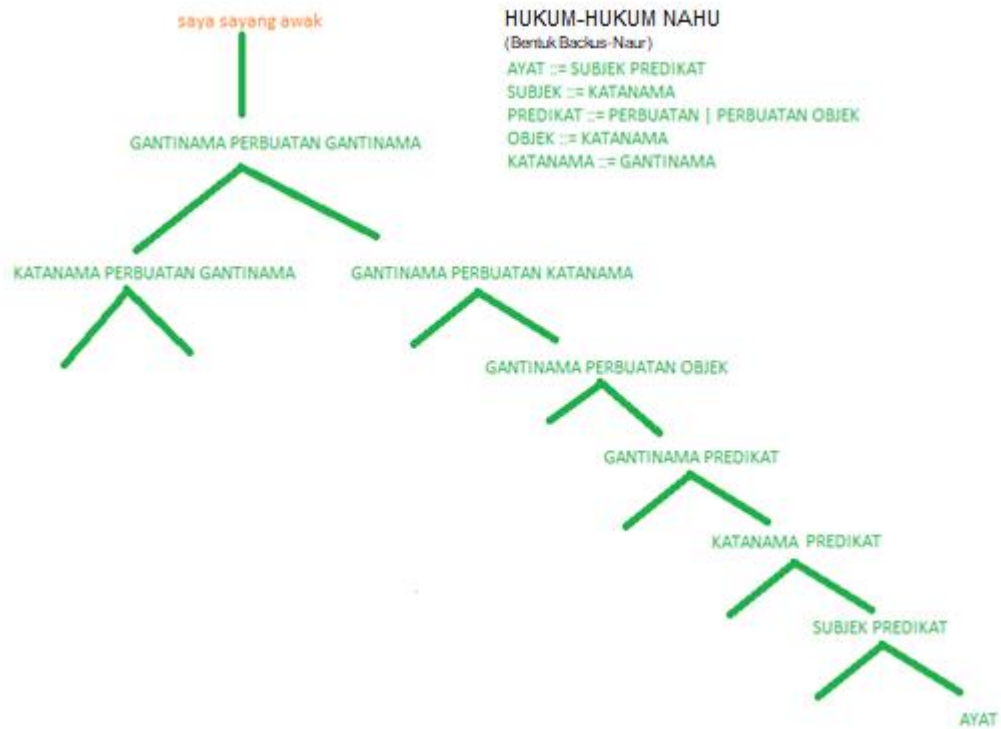
8.0.5 Bahasa

Dalam penyelesaian bahasa, apa yang diperlukan ialah pembuktian sesuatu rentetan itu merupakan ayat (jujukan kata-kata) sah menurut hukum-hukum nahu bahasa tersebut.

Hukum nahu berbentuk hukum sintaks bagi kata-kata, dan proses penghuraian dalam pembuktian bahasa adalah proses analisis sintaks. Token-token dikumpulkan ke dalam kenyataan-kenyataan menurut hukum-hukum nahu bagi bahasa berkenaan, dengan itu mengenalpasti peranan tiap-tiap komponen. Ini menghasilkan pepohon hurai.

Kumpulan token sah diubah kepada kumpulan token sah yang lain menerusi gerakan yang diberikan hukum nahu. Gelintaran dibuat bagi menunjukkan rentetan kata-kata sasaran itu terbitan sah daripada definisi ayat yang sah.

Di bawah ini ditunjukkan suatu contoh penghuraian, menggunakan gelintaran dipacu data. Bermula dengan “saya saying awak”, hukum-hukum nahu digunakan untuk menunjukkan bahawa rentetan ini boleh dibuktikan ‘AYAT’.

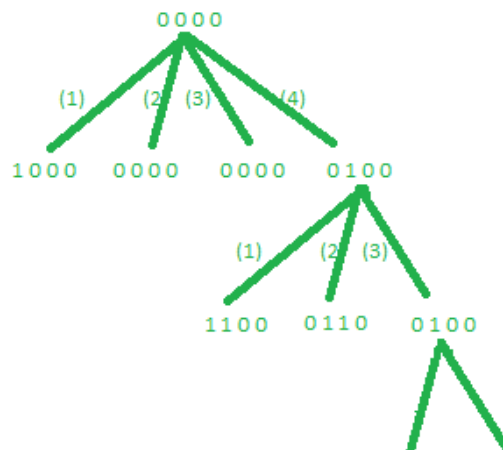


8.0.6 Logik

Masalah dalam logik pula ialah untuk mencari tafsiran logik yang memuaskan klausa logik yang diberi. Dalam logik usulan, mahu dicari nilai-nilai logik bagi usulan-usulan yang membuat klausa logik itu benar.

Ini diselesaikan menerusi penggelintaran mencari penyelesaian ini. Keadaan-keadaan dalam ruang masalah ialah nilai-nilai logik bagi usulan-usulan, dan gerakan diberi oleh hukum-hukum logik.

Contoh, diberi usulan-usulan p, q, r, s , dan hukum-hukum (1) p , (2) $q \rightarrow r$, (3) $(p \wedge r) \rightarrow s$, (4) q . Kalau dimula dengan 0,0,0,0, suatu pepohon gelintaran berbentuk seperti di bawah diperoleh.



8.0.7 Permainan

Permainan juga boleh dilihat sebagai penggelintaran – mencari keadaan yang memenangkan.

Untuk permainan satu-pemain, ini jelas. Keadaan dalam permainan, misalnya kedudukan buah-buah catur atas papan catur atau lain-lain, menanggung rangkaian masalah, dan gerakan diberikan oleh pergerakan dalam permainan, misalnya. pergerakan buah-buah catur seperti yang dibenarkan oleh peraturan permainan.

Bagi permainan dengan dua pemain, pergerakan adalah bergilir-gilir di antara dua pemain ini. Pada perspektif seorang pemain, dia mahu maksimumkan heuristiknya dalam gilirannya, dan dia jangkakan lawannya akan melakukan gerakan yang menguntungkan, iaitu yang punyai heuristik minimum untuk pemain pertama tadi. Jadi gelintaran merupakan jujukan memaksimumkan heuristik dan meminimumkan heuristik, atau disebut gelintaran minimaks.

Dalam permainan, biasanya keadaan-keadaan sangat banyak (bayangkan papan catur!), dan rangkaian masalah sangat besar untuk ditangani. Kadang-kadang pepohon gelintaran dicantas, dan pergerakan untuk hanya beberapa langkah dikira. Ini disebut ufuk pandang ke hadapan terhad.

8.1 Manipulasi Simbolan

8.1.0 Algebra

Algebra mengkaji simbol dan peraturan untuk memanipulasinya.

Kalau berkenaan nombor, ia melibatkan penggunaan pembolehubah untuk mewakili nombor, dan hubungan antara nombor tersebut.

8.1.1 Manipulasi simbolan berkomputer

Manipulasi simbolan mencari pengolahan algebra yang memberikan penyelesaian dicari, misalnya nilai pembolehubah, atau bentuk ungkapan dimudahkan. Keadaan dalam masalah diberi oleh kenyataan algebra. Pergerakan diberi oleh hukum-hukum algebra. Penggelintaran berheuristik dibuat.

Mekanismenya adalah seperti berikut:

Setiap langkah algebra → pergerakan dalam ruang keadaan

Pencarian ungkapan penyelesaian → gelintaran ruang keadaan

Heuristik algebra → heuristik gelintaran

Misalan perisian yang ada tersedia: Mathematica, MATLAB, MACSYMA, REDUCE, SHEEP, Maple, dan lain-lain.

Di antara kebolehan yang boleh didapati adalah:

- kembangkan ungkapan
- mudahkan ungkapan
- kembangkan sebagai siri
- selesaikan persamaan secara simbolan
- kamiran simbolan
- pembezaan simbolan
- selesaikan persamaan serentak

Juga, kadang-kadang ada digabungkan modul pengiraan berangka dan grafik bersama dengan kemudahan manipulasi simbolan.

Berikut adalah suatu contoh menggunakan MATLAB:

```
>>% selesaikan persamaan utk x:  
>>x = solve('a*x^2 + b*x + c = 0');  
>>x  
  
x =  
  
[ 1/2/a*(-b+(b^2-4*a*c)^(1/2))]
```

```
[ 1/2/a*(-b-(b^2-4*a*c)^(1/2))]
```

```
>>% selesaikan utk 2 pembolehubah:
```

```
>>[u,v] = solve('a*u^2 + v^2 = 0','u - v = 1')
```

```
u =
```

```
[ 1/2/(a+1)*(-2*a+2*(-a)^(1/2))+1]
```

```
[ 1/2/(a+1)*(-2*a-2*(-a)^(1/2))+1]
```

```
v =
```

```
[ 1/2/(a+1)*(-2*a+2*(-a)^(1/2))]
```

```
[ 1/2/(a+1)*(-2*a-2*(-a)^(1/2))]
```

```
>>% "D" menandakan pembezaan
```

```
>>y = dsolve('Dy = -a*y')
```

```
y =
```

```
C1*exp(-a*t)
```

```
>>% pembezaan kedua ditandakan "D2"
```

```
>>y = dsolve('D2y = -a^2*y', 'y(0) = 1, Dy(pi/a) = 0')
```

```
y =
```

```
cos(a*t)
```

8.2 Deduksi dan Keputusan

8.2.0 Deduksi

Deduksi, atau taakluan deduksian, bermaksud membuat taabiran berdasarkan fakta dan premis yang diterima. Ini adalah suatu proses logik.

8.2.1 Logik sebagai sistem formal

Logik mengkaji prinsip-prinsip penaakulan yang sah, dan menentukan kaedah untuk menilai sesuatu hujah atau pernyataan adalah benar atau salah berdasarkan peraturan tertentu.

Logik boleh dilihat sebagai suatu sistem formal.

Deduksi, atau taakluan deduksian, bermaksud membuat taabiran berdasarkan fakta dan premis yang

8.2.1.0 Sistem formal

Sistem formal memerihai sintaks suatu set objek data niskala. Ia mendefinisikan hukum-hukum sintaks dengan mana-mana set simbol niskala yang boleh dimanipulasi menggunakan hukum-hukum ini.

Sistem formal mengandungi:

1. Suatu ‘abjad’ (boleh berbentuk perkataan) terhad simbol-simbol (disebut “perbehendaraan kata”)
2. Tatacara untuk membentuk ‘perkataan’ daripada abjad ini (disebut “nahu”)
3. Suatu set aksiom-aksiom (semuanya perkataan)
4. Suatu set terhad hukum deduksi (dengan mana suatu set perkataan-perkataan baharu boleh dideduksi daripada suatu set diberikan),
$$u_1 \& u_2 \& \dots \& u_p \rightarrow w_1 \& w_2 \& \dots \& w_n$$

Maka boleh diadakan jujukan terhad perkataan-perkataan

$M_1, M_2, \dots, M_r$ di mana M_i adalah samada aksiom atau $M_j \rightarrow M_i, j < i$

Ini disebut “bukti”. “Teorem” adalah perkataan t yg mana wujud bukti dengan $M_r \equiv t$. Ini bermakna semua aksiom adalah teorem.

Kemudian, konsep “kebolehputusan” merujuk kepada keadaan di mana diberi suatu perkataan, ia boleh dibuktikan teorem atau dibuktikan bukan teorem (dengan langkah terhad) oleh sistem formal berkenaan.

Perhatikan bahawa setakat ini, formalisme sistem formal adalah berbentuk sintaksan, iaitu ia hanya melihat bentuk simbol dan gabungannya. Ia belum berkata apa-apa tentang makna simbol dan gabungan makna-makna. Ini ditangani oleh “tafsiran”, yang melihat bahagian semantik ini. Tafsiran menentukan hubungan di antara sistem formal dan dunia luar dengan memberi makna kepada

simbol-simbol. Dalam logik, simbol-simbol diberi nilai benar atau palsu, ditentukan oleh tafsiran. Mahu dicari tafsiran yang mana setiap teorem itu ayat yang benar. Ini dilakukan menerusi gelintaran.

Suatu sistem logik itu dikatakan sempurna jika semua perkataan sah (iaitu semua perkataan yang mempunyai nilai benar), atau semua ayat benar, menurut tafsiran semantik, itu merupakan teorem dari segi sintaks formal. Ia pula dikatakan betul jika semua teorem itu sah benar. Yang diingini ialah sistem logik yang sempurna dan betul.

8.2.1.1 Logik usulan

Logik usulan ialah sistem formal dengan:

- abjad yang terdiri daripada huruf-huruf usulan $p, q, r, \dots$, operator-operator logik $\neg$ ('bukan'), $\rightarrow$ (atau $\supset$) ('babatkan'), dan kurungan $(,)$
- perkataan yang terdiri daripada mana-mana huruf usulan, dan, jika w perkataan, (w) juga perkataan, jika w perkataan, $\neg w$ juga perkataan, jika w_1, w_2 perkataan, $w_1 \supset w_2$ juga perkataan
- aksiom, iaitu
$$A1: (w_1 \supset (w_2 \supset w_1))$$
$$A2: (w_1 \supset (w_2 \supset w_3)) \supset ((w_1 \supset w_2) \supset (w_1 \supset w_3))$$
$$A3: (\neg w_2 \supset \neg w_1) \supset (w_1 \supset w_2)$$
- hukum, iaitu hukum pemisahan / modus ponens :
$$(w_1) \ \& \ (w_1 \supset w_2) \rightarrow (w_2)$$

Operator-operator lain boleh didefinisikan:

DAN, $(w_1 \wedge w_2)$ yang sama dengan $\neg (w_1 \supset \neg w_2)$

ATAU, $(w_1 \vee w_2)$ yang sama dengan $\neg (\neg w_1 \wedge \neg w_2)$

IDENTITI, $(w_1 \equiv w_2)$ yang sama dengan $(w_1 \supset w_2) \wedge (w_2 \supset w_1)$

Tafsiran adalah berdasarkan pemberian nilai daripada $\{0,1\}$, di mana 0 ialah 'PALSU', dan 1 ialah 'BENAR', dan fungsi pelengkap $C(x)=1-x$, dan fungsi kebenaran $T(x,y)=0$ jika $(x=1, y=0)$, 1 jika tidak. Ini memberikan algebra Boolean yang sepadan dengan logik tabii.

Didapati,

- Logik usulan tak bertentangan (yakni F dan $\neg F$ tidak boleh dibuktikan serentak)
- Logik usulan sempurna
- Logik usulan betul
- Logik usulan bolehputus

8.2.1.2 Kalkulus predikat

Logik peringkat pertama membenarkan penggunaan pembolehubah dalam ayat-ayat, membenarkan ungkapan yang lebih khusus. Huruf-huruf usulan digantikan dengan predikat $A(x_1, x_2, \dots, x_k)$ dengan sebilangan pembolehubah. Bilangan perubahan di katakan ariti, di sini k .

Sistem formal bagi kalkulus predikat mengandungi:

- abjad: iaitu pemalar; fungsi; pembolehubah; predikat; pengkuantiti semesta $\forall$ ('bagi semua')
- perkataan: seperti untuk logik usulan ditambah predikat dan pengkuantiti semesta
- aksiom: seperti untuk logik usulan ditambah dengan

A4: $((\forall t)B(t)) \supset B(u)$

A5: $((\forall t)(w_1 \supset w_2)) \supset (w_1 \supset (\forall t)w_2)$ dengan t bukan pembolehubah bebas dalam w_1

hukum: iaitu modus ponens dan $w_1 \rightarrow ((\forall t)w_1)$ dengan t pembolehubah bebas dalam w_1 (itlakan)

Pengkuantiti wujudan boleh didefinisikan seperti berikut, $\exists$: $(\exists x)B(x) \equiv \neg((\forall x) \neg B(x))$.

Kalkulus predikat merupakan logik tahap pertama sahaja kerana $\forall, \exists$ bertindak atas pembolehubah sahaja, dan bukan predikat itu sendiri.

Tafsiran yang boleh digunakan menggunakan ketetapan berikut. Pemalar, pembolehubah, dan fungsi, yang boleh menjadi argumen dalam predikat, adalah ahli daripada set domain D . Fungsi di sini juga boleh punyai argumen, iaitu pemalar atau pembolehubah daripada D , dan memulangkan nilai daripada D juga. Predikat boleh punyai nilai daripada $\{\text{BENAR, PALSU}\}$, bergantung kepada argumennya. Pengkuantiti semesta $\forall$ menandakan kebenaran untuk semua nilai pemboleubah (diandaikan yang dalam D), sementara pengkuantiti wujudan $\exists$ memberi benar jika predikat berkenaan benar untuk mana-mana nilai pembolehubah.

Kerana penyelesaian kalkulus predikat membabitkan gelintaran terhadap ahli-ahli D , jika D tak terhingga maka kalkulus predikat takbolehputus.

8.2.1.3 Resolusi

Resolusi merupakan suatu kaedah untuk penyelesaian sesuatu sistem logik. Gelintaran dipacu sasaran dipakai, menggunakan *reductio ad absurdum* menerusi *modus tollens*.

Dicari tafsiran (pemetaan usulan/predikat ke nilai-nilai kebenaran) yang sah. Bagi logik usulan, ini membabitkan penentuan samada sesuatu perkataan itu teorem atau bernilai benar (kerana ia sempurna dan betul), atau sebaliknya.

Kalau S mahu ditunjukkan benar, menggunakan *reductio ad absurdum* tadi, keadaan sasaran dicari adalah di mana $\neg S$ tak benar. Gerakan dalam gelintaran menggunakan *modus tollens*, iaitu

$$\neg Q \ \& \ (P \supset Q) \rightarrow \neg P$$

Lihat contoh ini.

- Diberi (1) p
- (2) $q \supset r$
- (3) $(p \wedge r) \supset s$
- (4) q

Tunjukkan s .

Penyelesaian:

- Katakan $\neg s$ benar
- $\Rightarrow \neg(p \wedge r)$ drp (3), *modus tollens*
- $\Rightarrow \neg p \vee \neg r$
- $\Rightarrow \neg r$ drp (1)
- $\Rightarrow \neg q$ drp (2), *modus tollens*
- $\square$ mencanggahi (4)
- maka s benar menerusi *reductio ad absurdum*

Untuk kalkulus predikat, diperlukan juga mekanisme penyatuan untuk menangani pembolehubah. Padanan diperolehi, bagi predikat-predikat sama, dengan gantian pembolehubah yang paling umum supaya ungkapan kalkulus predikat padan. Contoh,

$\text{foo}(\text{X}, \text{a}, \text{goo}(\text{Y}))$	padan dengan	$\text{foo}(\text{fred}, \text{a}, \text{goo}(\text{Z}))$
		dengan ikatan $\{\text{X}/\text{fred}, \text{Z}/\text{Y}\}$
	padan dengan	$\text{foo}(\text{W}, \text{Z}, \text{goo}(\text{jack}))$
		dengan ikatan $\{\text{X}/\text{W}, \text{Z}/\text{a}, \text{Y}/\text{jack}\}$
	padan dengan	$\text{foo}(\text{Z}, \text{a}, \text{goo}(\text{moo}(\text{Z})))$
		dengan ikatan $\{\text{Z}/\text{X}, \text{Y}/\text{moo}(\text{Z})\}$
	tak padan dengan	$\text{foo}(\text{fred}, \text{b}, \text{goo}(\text{Z}))$

dan seterusnya

Berikut adalah suatu contoh kegunaan resolusi dalam suatu masalah kalkulus predikat.

contoh:

diberi: (1) $\text{anak}(\text{ahmad}, \text{abdullah})$
 (2) $\text{anak}(\text{abdullah}, \text{sulaiman})$
 (3) $(\forall X \forall Y \forall Z) \text{anak}(\text{X}, \text{Y}) \wedge \text{anak}(\text{Y}, \text{Z}) \supset \text{cucu}(\text{X}, \text{Z})$

sasaran: $\text{cucu}(\text{ahmad}, \text{sulaiman})?$

katakan $\neg \text{cucu}(\text{ahmad}, \text{sulaiman})$

(bukti)

padan dgn cucu dlm (3) dgn $\{\text{X}/\text{ahmad}, \text{Z}/\text{sulaiman}\}$

$\Rightarrow \neg (\forall Y)(\text{anak}(\text{ahmad}, Y) \wedge \text{anak}(Y, \text{sulaiman}))$

$\Rightarrow \neg (\forall Y)(\text{anak}(\text{ahmad}, Y)) \wedge \neg (\forall Y)(\text{anak}(Y, \text{sulaiman}))$

padan dgn (1) dgn $\{\dots\} \cup \{Y/\text{abdullah}\}$ gabung dgn ikatan terdahulu

$\Rightarrow \neg \text{anak}(\text{abdullah}, \text{sulaiman})$

padan dgn (2)

□ percanggahan

$\Rightarrow \text{cucu}(\text{ahmad}, \text{sulaiman})$ benar

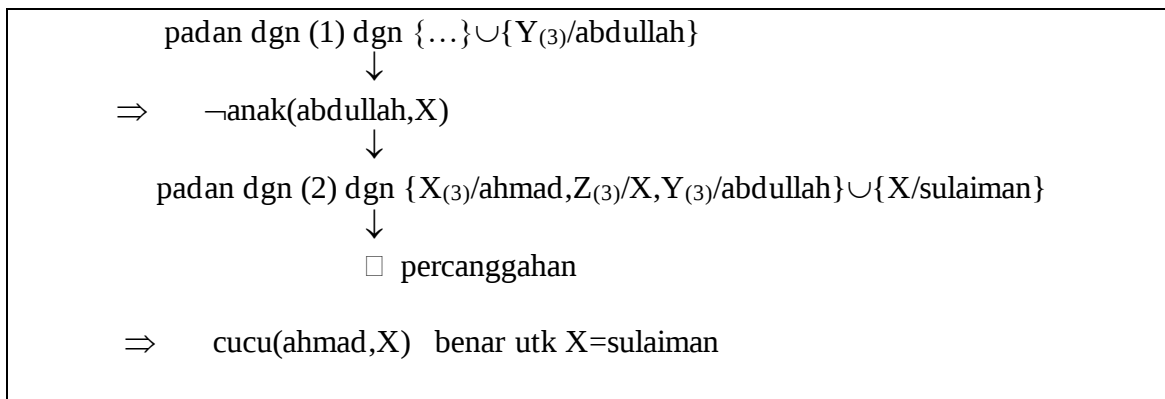
juga, pencarian: $\text{cucu}(\text{ahmad}, X)?$

katakan $\neg \text{cucu}(\text{ahmad}, X)$

padan dgn cucu dlm (3) dgn $\{\text{X}_{(3)}/\text{ahmad}, \text{Z}_{(3)}/X\}$

$\Rightarrow \neg (\forall Y_{(3)})(\text{anak}(\text{ahmad}, Y_{(3)}) \wedge \text{anak}(Y_{(3)}, X))$

$\Rightarrow \neg (\forall Y_{(3)})(\text{anak}(\text{ahmad}, Y_{(3)})) \vee \neg (\forall Y_{(3)})(\text{anak}(Y_{(3)}, X))$



8.2.2 Pengaturcaraan logik

Kerangka kalkulus predikat dan resolusi membolehkan pengaturcaraan logik. Pengaturcaraan logik menggunakan klausa-klausa logik isytiharan sebagai aturcara, dan resolusi digunakan untuk menjalankan aturcara itu.

Gunakan subset kalkulus predikat yang membolehkan ini – iaitu klausa-klausa logik dalam bentuk klausa Horn. Ini adalah klausa-klausa ‘jika’ dengan tidak lebih daripada satu akibat.:

$$A(x_1, x_2, \dots, x_k) \leftarrow B_1(\dots), B_2(\dots), \dots, B_n(\dots).$$

hanya satu kepala $\subset$ $\wedge$ $\wedge$ $\wedge$

‘hukum’

dibaca $A(x_1, x_2, \dots, x_k)$ jika $B_1(\dots)$ dan $B_2(\dots)$ dan ... dan $B_n(\dots)$. Ini disebut suatu ‘hukum’. Jika jika n sifar, ia dikatakan penegasan, yakni $A(\dots)$ benar. Jika tiada kepala, ia merupakan penafian, yakni antejadian palsu.

Gunakan resolusi, dengan $\forall$ diandaikan utk pembolehkan. Pengkuantiti wujudan $\exists$ ditangani oleh pengskoleman di mana suatu fungsi (fungsi Skolem) digunakan bagi pembolehkan berkenaan, misalnya

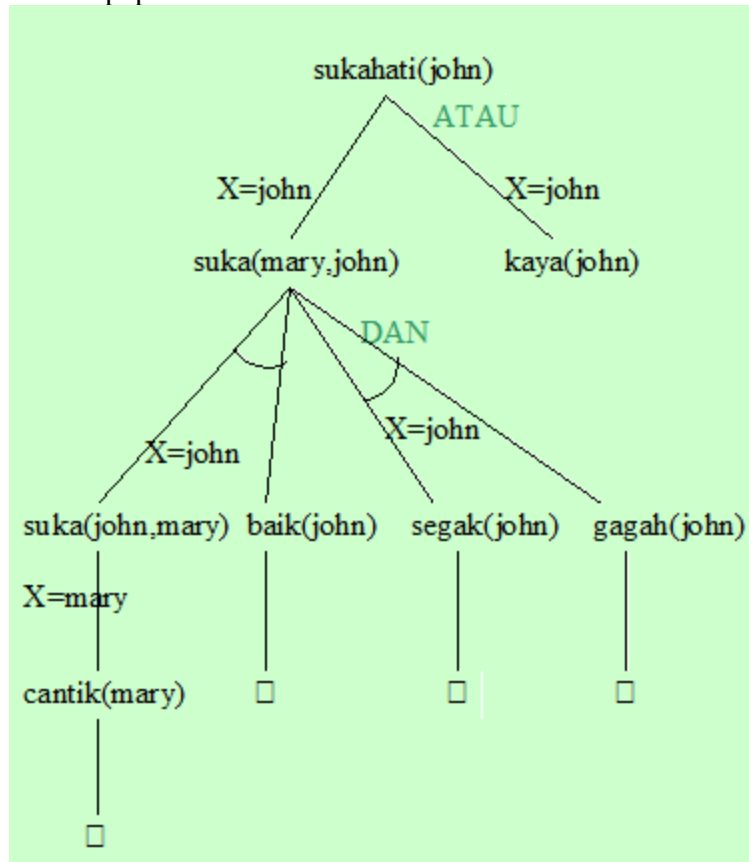
$$(\forall X)(\exists Y)(ibu(X, Y)) \text{ jadi } ibu(X, mak(X))$$

Gelintaran ruang bukti dilakukan. Sasaran digantikan subsasaran dengan dahan ‘DAN’ apabila kepala dipadankan dan digantikan dengan antejadian. Ada kemungkinan perlu dicuba beberapa hukum dengan kepala serupa, memberikan dahan ‘ATAU’. Juga, mungkin perlu dicuba beberapa gantian (ikatan).

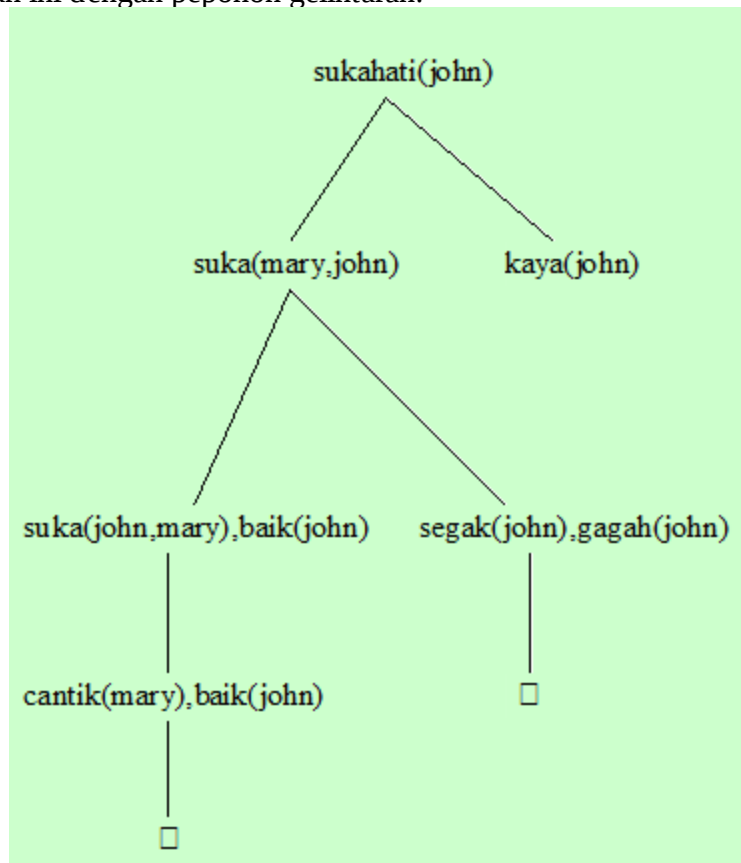
Berikut ini diberikan suatu contoh.

sasaran: $\leftarrow gembira(john).$ hukum: $sukahati(X) \leftarrow kaya(X).$ $sukahati(X) \leftarrow suka(mary, X).$ $suka(mary, X) \leftarrow suka(X, mary), baik(X).$ $suka(mary, X) \leftarrow segak(X), strong(X).$ $suka(john, X) \leftarrow cantik(X).$ penegasan: $cantik(mary) \leftarrow.$ $baik(john) \leftarrow.$ $segak(john) \leftarrow.$ $gagak(john) \leftarrow.$

Ini memberikan pepohonan dan-atau berikut..



Bandingkan ini dengan pepohonan gelintaran:



Suatu aturcara logik terdiri daripada penegasan-penegasan, hukum-hukum, dan sasaran. Penegasan dan hukum dikatakan pangkalan pengetahuan. Gaya pengaturcaraannya disebut pengaturcaraan isytiharan berbanding pengaturcaraan tatacaraan biasa.

Klausa hukum juga boleh ditafsirkan secara tatacaraan, di mana predikat dalam kepala klausa memanggil predikat-predikat dalam ekor klausa sebagai subfungsi. Misalnya, berikut ialah aturcara untuk mengira nilai faktorial:

```
faktorial(0,1) ← .
```

```
faktorial(X,Y) ← faktorial(X1,Y1), darab(X,Y1,Y), tolak(X,1,X1) .
```

Fungsi 'darab' dan 'tolak' di sini merupakan fungsi-fungsi tersedia terbina-dalam.

8.2.3 Sistem pakar

Sistem pakar adalah sistem taakulan dengan domain terhad. Ia biasanya terdiri daripada suatu pangkalan pengetahuan, dan suatu enjin taabiran yang menggerakkan taakulan. Pengetahuan dan mekanisme taabiran ini mungkin berasingan dan enjin taabiran yang sama boleh digunakan ke atas pangkalan pengetahuan yang lain juga.

Pengetahuan diperolehi daripada pakar sebenar dalam apa yang dipanggil proses pemerolehan pengetahuan. Taabiran dilakukan menggunakan asas logik, dan juga boleh mengandungi ketidakpastian. Boleh juga diadakan mekanisme penjelasan untuk menerangkan aliran taabiran. Output sistem pakar biasanya adalah dalam bentuk nasihat.

Di sini adalah suatu contoh daripada sistem pakar MYCIN:

Contoh dari MYCIN (1976)

hukum...

RULE 85: IF location of culture is blood & organism is gram-negative & organism is long in shape & patient is endangered host THEN it is probable (0.6) that the organism is *pseudomonas aeruginosa*.

...

RULE 217: IF organism is a bacteroid & location of culture is sterile THEN therapy should be chosen from the following: *chloramphenicol*, *clindamycine*, *tetracycline*, *lindomycine*, *gentamycine*.

metahukum...

METARULE 2: IF patient is an endangered host & there is rule which mentions *pseudomonas* in its premise & there is rule which mentions *klebsiellas* in its premise THEN with probability 0.4 use the first before the second.

...

METARULE 25: IF location of culture is not sterile & there is rule which mentions in its premise the organism found in the patient and which is possibly the same that is sought THEN it is probable (probability 1) that none of these rules are useful.

Interaksi...

@mycin

mycin 26-mac-96

instructions? (Y or N)

N

26_march 10:42

--- PATIENT-1 ---

ingatan bekerja - dinamik

(1) Name of patient:

pangkalan pengetahuan

Sammy Welly

(2) Age?

34

(3) Sex?

why (kenapa soalan ini ditanya)

[i.e., Why is it important to know the sex of PATIENT-1?]

[1.0] The sex of a patient is one of the four standard parameters to know the patient. This may be useful later on...

[back to question 3 ...]

male

(4) Race?

African

(5) Does there exist any cultures for Sammy Welly which may be related to the present illness, and from which organisms have been successfully produced in the lab?

no

(6) Does there exist any culture for Sammy Welly which is waiting?

yes

--- CULTURE-1 ---

(7) From which location is the specimen for CULTURE-1 obtained?

CSF

(8) Please give date and time when culture CSF was obtained?

26-Mar-7:00

(9) Was the smear examined when the specimen was taken?

ya

(10) Has any culture been seen at the specimen smear?

no

(11) Are there any more cultures waiting?

no

...

(16) Does Sammy Welly show symptoms of extended headaches or other abnormal neurological symptoms (headache, lethargy, etc) recently?

yes

(17) Has Sammy Welly got objective proof for abnormal neurological signs (facial tightness, coma, attacks, etc) recently which was recorded by a doctor?

yes

(cuba meningitis – rangkaian ke belakang utk mengenal organisme)

Culture CSF will be considered as related to MENINGITIS.

(18) Please give date when clinical proof (symptom, signs or laboratory test) for meningitis first obtained.

25 March 10:00

...

(36)Has Sammy Welly undergone neurosurgery?

why

[i.e., Why is it important to determine whether PATIENT-1 has undergone neurosurgery?]

[6.0] This will help in determining the organism (other than that seen in culture or smear) which may cause the infection.

It has been established that

[6.1] culture which requires therapy is meningitis, and

[6.2] organism is not seen in culture CSF (CULTURE-1), and

[6.3] infection type is bacterial

therefore, if

[6.4] Sammy Welly has undergone neurosurgery, and

8.3 Induksi dan Pembelajaran

8.3.0 Induksi

Kita kaitkan 2 proses seakan logik dengan induksi atau aruhan, iaitu itlakan dan kiasan.

Dalam itlakan, rumusan dibuat tentang yang am berdasarkan yang khusus. Contoh, angsa yang pernah dilihat, dan dalam banyak bilangannya, semua berwarna putih. Diitlakkan bahawa semua angsa putih. Ia tidak kukuh logik sebenarnya, dan runtuh bila angsa hitam misalnya dilihat.

Dalam kiasan, atau analogi, pula, kerana dua benda ada keserupaan, apa yang benar untuk yang pertama dikiasan benar untuk yang kedua. Contoh, Awang dan Embong kedua-dua berasal Kelantan, buat kursus fizik dan tinggal di Kolej Ketiga. Awang suka budu. Embong dijangkakan suka budu juga. Rumusan beranalogi ini juga tidak kukuh logik.

Induksi tidak kukuh logik. Ia bukan teorem dalam sistem formal.

8.3.1 Pembelajaran

Pembelajaran menghasilkan penambahan pengetahuan. Ia boleh datang daripada itlakan dan kiasan.

Kesan pembelajaran adalah,

- ia membawa kepada mencakupi masalah-masalah dengan lebih luas (dengan keadaan baharu; berlaku itlakan)
- pemberian jawapan-jawapan lebih tepat
- menyebabkan pencapaian jawapan dengan kos lebih rendah
- meringkaskan pengetahuan terstor

Boleh dikaitkan tahap-tahap kaedah pembelajaran seperti berikut, secara meingkat,

1. diaturcarakan

Pengetahuan, terutama pengetahuan tatacraan, diaturcarakan secara keras.

2. penghafalan

Stor segenap keadaan mungkin, bersama tindakan berkenaan setiapnya.

3. statistikan

Ini seperti 2. tetapi ia menggunakan statistik. Timbul kelas-kelas.

4. pembelajaran dari misalan

Diberi misalan-misalan, diitlakkan keadaan-keadaan baharu. Ini juga disebut pembelaran diselia atau pembelajaran dengan guru, kerana misalan-misalan benar ditunjukkan.

5. pembelajaran menerusi penemuan

Itlak keadaan yang ada, membawa kepada hipotesis atau konsep baharu. Ini juga disebut pembelaran biselia atau pembelajaran tanpa guru.

Dalam pembelajaran, apa yang dipelajari bolehjadi – parameter, jejak gelintaran, hukum, rencana, ciri, konsep, heuristik, dan lain-lain lagi.

Ada beberapa kaedah pembelajaran yang telah dicadangkan. Berikut diterangkan di antaranya.

8.3.1.0 Gelintaran

Gelintar ruang konsep dan sebagainya yang mungkin.

Misalnya, gelintar ruang hukum-hukum yang mungkin yang memerihalkan data yang diberi menerusi misalan, misalan tentangan dan hampir kena. Sertakan misalan ke dalam konsep menerusi penyeluruhan hukum, dan singkirkan misalan tentangan dan terutamanya hampir kena menerusi pengkhususan hukum

8.3.1.1 Pepohon keputusan

Bina pepohon keputusan daripada data. Guna sebahagian data ('tingkap') untuk membina pepohon paling mudah dan cuba pepohon ini untuk semua data lain. Besarkan tingkap jika pepohon tidak memerihalkan semua data.

8.3.1.2 Pembelajaran berdasarkan penerangan

Cuba terangkan misalan, dengan itu memperoleh peneluran. 'Penerangan' ialah bagaimana misalan adalah pengsekitakan konsep sasaran. Cantas aspek-aspek tak penting dalam misalan menggunakan pengetahuan domain.

8.3.1.3 Penemuan

Ini merupakan kaedah pembelajaran tanpa guru. Gunakan heuristik dipacu teori atau dipacu data. Juga, boleh digunakan penggugusan konsep: ada ukuran keserupaan antara objek, dan guna algoritma penggugusan. Memberikan taksonomi.

8.3.1.4 Analogi

Analogi transformasian dilakukan. Selesakani masalah baharu dengan ubahsuaian kaedah penyelesaian-penyelesaian lama.

8.3.1.5 Neuronan

Guna rangkaian neuronan, misalnya rangkaian neuronan berlapisan suap hadapan dengan rambatan balik ralat. Neuron daripada satu lapisan menentukan keaktifan neuron dalam lapisan seterusnya, menerusi gabungan masukan daripada sambungan sinaps berlainan kekuatan T_{ij} ,

$$V_i = g(h_i(t))$$

di mana

$$h_i = \sum_j T_{ij}V_j$$

dengan fungsi hantaran g berbentuk sigmoid seperti misalnya fungsi ralat. Bagi setiap misalan masukan keluaran, T_{ij} disesuaikan untuk meminimumkan ralat, iaitu perbezaan di antara nilai neuron didapati berbanding yang dikehendaki, dalam lapisan terakhir (lapisan keluaran),

$$\varepsilon = \frac{1}{2} \sum_{i \in L_{\text{kel}}} (V_i - \xi_i)^2,$$

menggunakan kaedah Newton-Ralphson,

$$\Delta T_{ij} = -\alpha \frac{\partial \varepsilon}{\partial T_{ij}}.$$

8.3.1.6 Genan

Guna algoritma genan yang meniru evolusi menerusi prinsip kemandirian yang tercocok. Sediakan populasi calon-calon penyelesaian. Lakukan perubahan ke atas calon-calon ini menerusi operator mutasi ke atas individu dan pindahsilang di antara individu. Pilih ahli-ahli populasi yang terbaik (tercocok) dan ulang tatacara, sehingga calon terbaik diperolehi.

8.4 Penjanaan

8.4.0 Janaan

Kepintaran buatan janaan ialah kaedah yang mampu menjana teks, imej atau media lain, menggunakan model janaan.

Model janaan, dari segi taburan kebarangkalian, di bina dengan dipelajari corak dan struktur data latihan input. Model ini kemudian digunakan menjana data baharu yang mempunyai ciri yang serupa.

8.4.1 Penjana

Antara lain, rangkaian neuron (dalam) berasaskan senibina transformer boleh digunakan.

Rangkaian neuronan janaan ini digunakan dalam Model Besar Bahasa (LLM) yang menghasilkan teks, dan Model Besar Tampak (LVM), yang menghasilkan imej, menerusi penjanaan berkebarangkalian yang dipelajari daripada data-data sampel masukan.

Tugasan 14

1. Tulis aturcara untuk melakukan gelintaran kealaman dahulu untuk mencari penyelesaian masalah jurujual jalanan seperti berikut:

Bermula dari A. Perlu dilawat setiap tempat sekali. Jadual di bawah memberikan jarak di antara tempat-tempat ini.

Beri penyelesaian (jujukan tempat, dan jarak jumlah) bagi dua kes diberikan.

Beri juga penyelesaian bagi penyelesaian menggunakan heuristik jiran terdekat.

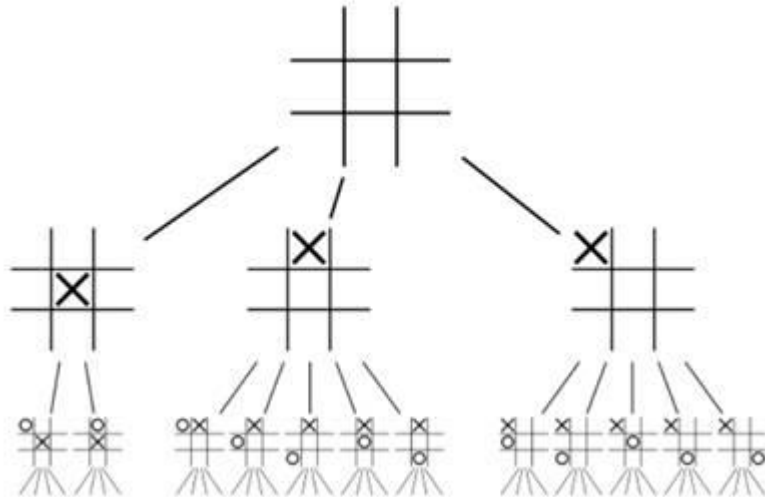
a)

	A	B	C	D	E	F	G
A		3.4	9.2	1.3	1.6	5.0	2.2
B	3.4		5.1	3.3	2.5	3.2	4.5
C	9.2	5.1		1.9	3.0	3.8	6.0
D	1.3	3.3	1.9		3.1	2.7	5.3
E	1.6	2.5	3.0	3.1		1.9	2.3
F	5.0	3.2	3.8	2.7	1.9		3.6
G	2.2	4.5	6.0	5.3	2.3	3.6	

b)

	A	B	C	D	E	F	G	H	I
A		1.6	5.0	2.2	3.2	3.8	2.7	1.9	1.6
B	1.6		4.5	6.0	5.3	2.3	1.6	3.0	2.5
C	5.0	4.5		7.5	3.8	3.2	2.5	3.8	3.0
D	2.2	6.0	7.5		6.0	3.8	3.0	6.0	3.1
E	3.2	5.3	3.8	6.0		2.7	3.1	1.3	1.9
F	3.8	2.3	3.2	3.8	2.7		4.0	3.3	3.0
G	2.7	1.6	2.5	3.0	3.1	4.0		1.9	3.8
H	1.9	3.0	3.8	6.0	1.3	3.3	1.9		6.0
I	1.6	2.5	3.0	3.1	1.9	3.0	3.8	6.0	

2. Pepohon permainan (pepohon gelintaran untuk permainan) untuk permainan kandang babi (*noughts and crosses*) adalah seperti berikut:



Tulis aturcara untuk menggelintar pergerakan-pergerakan permainan untuk bermainnya dengan lawan luar menggunakan pilihan minimaks melihat tiga langkah di hadapan (ufuk pertimbangan). Sebut pilihan fungsi heuristik yang anda gunakan. Cuba main lawan aturcara anda.

Tugas 15

1. Guna sistem aljabra komputer (mis. Maxima) untuk menyelesaikan masalah aljabra secara simbolan, termasuk yang membabitkan kamiran/pembezaan.
2. Tulis suatu pangkalan pengetahuan dalam bahasa logik Prolog (mis. Guna GNU Prolog) yang mengandungi hubungan keluarga anda untuk 3 generasi. Menggunakan ini, buat gesaan untuk menyenaraikan: (i) sepupu anda, dan (ii) ibu saudara anda.
3. Cuba sistem janaan bahasa tabii (mis. ChatGPT). Suruh ia menjana teks tertentu.