

Bahasan Terakhir...

- Pencarian Iteratif
 - Simulated Annealing
 - Pencarian Tabu
 - Mean Ends
- Pencarian Adversarial
 - Minimax (Min-Max)
 - Alpha-Beta Pruning

Tugas Hard Copy (Lanjutan...)

- Pencarian Iteratif
 - **Simulated Annealing**
 - **Pencarian Tabu**
- Pencarian Adversarial
 - **Minimax (Min-Max)**
 - **Alpha-Beta Pruning**
- Sebutkan definisi dan Algoritmanya. Berikan contoh disertai langkah-langkah penyelesaian.
- Dikumpulkan saat UTS

Ujian Tengah Semester (UTS)

- Materi Ujian:
 - **Simulated Annealing** atau **Pencarian Tabu**
 - **Penyelesaian Masalah dengan *First Order Logic***
- **Sifat Ujian: Terbuka, Individu.**

Kecerdasan Buatan

Pertemuan 06

Representasi Pengetahuan & Penalaran

Kelas 10-S1TI-03, 04, 05

Husni

Lunix96@gmail.com

<http://Komputasi.wordpress.com>

Outline

- Pendahuluan
- Logika Proposisi (Propositional Logic, PL)
- First Order Predicate Logic (FODL)
- Resolusi dalam PL dan FODL
- Rangkuman
- Tugas

Pendahuluan

- Pengetahuan
 - Informasi dari lingkungan yang dapat direpresentasikan dalam bentuk proposisi.
- Representasi Pengetahuan
 - Simbol yang digunakan untuk merepresentasikan proposisi tersebut.
- Representasi Pengetahuan & Penalaran
 - Manipulasi simbol-simbol yang meng-*encode*-kan proposisi untuk menghasilkan representasi dari proposisi baru.

Basis Pengetahuan

- Komponen utama dari sistem berbasis pengetahuan
- Himpunan kalimat yang diekspresikan dalam suatu bahasa (bahasa representasi pengetahuan)
- Kalimat menyajikan beberapa pernyataan mengenai dunia ini.
- Mekanisme menurunkan kalimat baru dari yang lama dikenal sebagai inferensi atau penalaran (*reasoning*)
- Syarat utama inferensi: kalimat baru harus mengikuti secara logis kalimat sebelumnya

Logika

- Suatu metode representasi (bahasa) yang sering digunakan di dalam AI
- Keuntungan: tepat, pasti, memungkinkan penalaran mengenai negatif dan disjungsi
- Deklaratif: Apa sesuatu itu benar atau salah, true atau false, 1 atau 0.
- Logika tidak dapat dengan baik merepresentasikan ketidakpastian (*uncertainty*)

Fakta & Representasi

- Fakta:
 - Penegasan mengenai dunia ini
 - Dapat bernilai True atau False.
- Representasi:
 - Ekspresi/kalimat dalam beberapa bahasa
 - Dapat di-*encode*-kan dalam program komputer
 - Mengandung obyek dan relasinya
 - Harus konsisten dengan kenyataan.

2 Bagian Logika

- **Bahasa**, terdiri dari dua aspek:
 - **Sintaks**. Menyajikan simbol *atomic* dari bahasa logis dan aturan untuk membangun ekspresi *non-atomic* (struktur simbol)-nya. Menentukan simbol dalam bahasa dan bagaimana dapat dikombinasikan untuk membentuk kalimat. Fakta disajikan sebagai kalimat.
 - **Semantik**. Memberikan makna untuk simbol atomic. Menentukan fakta apa yang dirujuk suatu kalimat. Juga menentukan bagaimana memberikan nilai kebenaran ke suatu kalimat berdasarkan pada maknanya dalam dunia ini.
- **Metode Penalaran**. Terdiri dari aturan-aturan untuk menentukan subset dari ekspresi logis, disebut teorema logika. Mengacu ke metode mekanis untuk menurunkan kalimat baru dari kalimat yang ada.
- Ada sejumlah sistem logis dengan sintaks dan semantik berbeda.

Logika Proposisi

- Sistem logis paling sederhana
- Kita harus mendefinisikan sehimpunan simbol proposisi, misal: P, Q. Kemudian definisikan semantik dari simbol tersebut.
- Contoh:
 - P berarti “Minggu adalah hari libur”.**
 - Q berarti “Sekarang adalah hari minggu”.**
- Himpunan operator digunakan dalam proses penalaran terhadap nilai-nilai kebenaran.

Operator Logika

- Operator Dasar

- And, Dan \wedge
- Or, Atau \vee
- Not, Tidak \neg
- Implies, Maka, Menyebabkan \Rightarrow
- Iff (if and only if), Jika dan hanya jika \Leftrightarrow

- Contoh Logika Proposisi:

- R: Sekarang Hujan
- D: Sekarang Gelap
- C: Sekarang Dingin

Bahasa/Formula Didefinisikan:

- a) Simbol (suatu $\wedge, \vee, \neg, \Rightarrow, \Leftrightarrow$);
- b) Jika P kalimat, maka $\neg P$ juga suatu kalimat;
- c) Jika P dan Q kalimat, maka $P \wedge Q, P \vee Q, P \Rightarrow Q$ dan $P \Leftrightarrow Q$ juga kalimat;
- d) Kalimat yang mengandung aplikasi dari (a)-(c)

- Contoh:

$R \wedge C$ Sekarang hujan dan sekarang dingin.

$R \Rightarrow C$ Jika sekarang hujan maka sekarang dingin.

$(R \wedge D) \Rightarrow C$ Jika sekarang hujan dan sekarang gelap maka sekarang dingin.

Operator Not (\neg)

- Operator uner, hanya dapat diterapkan terhadap satu variabel.
- \neg true adalah false
- \neg false adalah true

P	\negP
true	false
false	true

Operator AND (\wedge)

- Operator biner, diberlakukan terhadap 2 variabel.
- Disebut juga operator konjuntif. $P \wedge Q$ adalah konjungsi dari P dan Q.

P	Q	$P \wedge Q$
true	true	true
true	false	false
false	true	false
false	false	false

Operator OR (\vee)

- Termasuk operator biner
- Disebut pula operator disjungtif. $P \vee Q$ adalah disjungsi dari P dan Q.

P	Q	$P \vee Q$
true	true	true
true	false	true
false	true	true
false	false	false

Operator Implikasi (\Rightarrow)

- Pada pernyataan $P \Rightarrow Q$, P adalah *antecedent*, dan Q *consequent*
- $P \Rightarrow Q$ dibaca: “ P mengakibatkan Q ” atau “jika P maka Q ” atau “jika P benar maka Q benar”.
- Jika P : “Saya suka coklat” dan Q : “Saya makan coklat”, maka jika P benar dan Q benar maka $P \Rightarrow Q$ benar (Jika saya suka coklat maka saya makan coklat).
- Jika P benar dan Q salah maka $P \Rightarrow Q$ juga salah (Jika saya suka coklat maka saya tidak makan coklat).
- Jika P dan Q salah maka $P \Rightarrow Q$ benar (Jika saya tidak suka coklat maka saya tidak makan coklat).

Operator Iff (\Leftrightarrow)

- Operator Iff (jika dan hanya jika) bernilai True hanya jika kedua variabel bernilai True atau False.

- Implikasi:

P	Q	$P \Rightarrow Q$
true	true	true
true	false	false
false	true	true
false	false	true

- Iff:

P	Q	$P \Leftrightarrow Q$
true	true	true
true	false	false
false	true	false
false	false	true

Kombinasi

P	Q	R	$\neg P$	$\neg P \wedge Q$	$(\neg P \wedge Q) \vee R$
true	true	True	false	false	true
true	true	False	false	false	false
true	false	False	false	false	false
true	false	True	false	false	true
false	true	True	true	true	true
false	true	False	true	true	true
false	false	True	true	false	true
false	false	False	true	false	false

Inferensi

- Kesimpulan dari sehimpunan asumsi diperoleh dengan memberlakukan beberapa aturan inferensi.
- Diberikan dua kalimat P dan Q. Dikatakan Q disimpulkan dari P ($P \vdash Q$) jika ada urutan aturan inferensi yang berlaku untuk P dan memungkinkan Q untuk ditambahkan.
- Ada beberapa aturan inferensi dalam logika proposisi.

Pemunculan (*Introduction*)

- **Pemunculan AND**

$$\frac{P \quad Q}{P \wedge Q}$$

Diberikan P dan Q, dapat disimpulkan $P \wedge Q$.

- **Pemunculan OR**

$$\frac{P}{P \vee Q}, \frac{Q}{P \vee Q}$$

Dari P dapat disimpulkan disjungsi dari P dengan suatu ekspresi ($P \vee Q$ benar untuk suatu nilai Q).

- **Pemunculan Implikasi**

Jika dimulai dari asumsi P dan diturunkan kesimpulan C, maka dapat disimpulkan bahwa $P \Rightarrow C$

$$\frac{P \quad \vdots \quad C}{P \Rightarrow C}$$

Eliminasi

- **Eliminasi AND**. Diberikan $P \wedge Q$, dapat diturunkan P dan Q secara terpisah.

$$\frac{P \wedge Q}{P} \quad \frac{P \wedge Q}{Q}$$

- **Eliminasi Implikasi (Modus Ponens)**. Jika P benar dan P mengakibatkan Q benar, maka dapat diketahui bahwa Q benar.

$$\frac{P \quad P \Rightarrow Q}{Q}$$

- Contoh:

P: Saya suka coklat. Q. Saya makan coklat.

Saya suka coklat. Jika saya suka coklat maka saya makan coklat. Jadi: **Saya makan coklat.**

- **Eliminasi NOT**. Jika suatu kalimat dinegasi dua kali diperoleh kalimat itu sendiri, tanpa negasi.

$$\frac{\neg \neg P}{P}$$

Resolusi

- **Resolusi Unit:**

$$\frac{\alpha \vee \beta, \neg \beta}{\alpha}$$

- **Resolusi:**

$$\frac{\alpha \vee \beta, \neg \beta \vee \gamma}{\alpha \vee \gamma} \text{ ekuivalen dengan } \frac{\neg \alpha \Rightarrow \beta, \beta \Rightarrow \gamma}{\neg \alpha \Rightarrow \gamma}$$

Contoh: Dunia Wumpus

- Monster bernama Wumpus tinggal di gua yang terbagi dalam 16 ruangan.
- Ada 3 lubang mematikan (*Pit*) yang mengeluarkan angin (*Breeze*) ke sekitarnya.
- Wumpus mengeluarkan bau busuk (*Stench*).
- Wumpus menjerit (*Scream*) saat mati terkena panah
- Jika agent menabrak dinding gua maka benjol (*Bump*)
- Agent harus memanah Wumpus atau mengambil *Gold*.

Contoh: Dunia Wumpus

4	Stench		Breeze	Pit
3	Wumpus Stench	Breeze Stench Gold (glitter)	Pit	Breeze
2	Stench		Breeze	
1	START Agent →	Breeze	Pit	Breeze
	1	2	3	4

Contoh: Dunia Wumpus

- Solusinya diselesaikan melalui 3 fungsi:
 - **Percept**, sesuatu yang ditangkap atau dirasakan, dapat berbentuk [stench, breeze, glitter, bump, scream]
misal: [stench, breeze, none, none, none]
 - **Action**, tindakan yang dikerjakan agent, misal: bergerak, ambil, panah, panjat
 - **Goal**, apakah tujuan tercapai?

Langkah-langkah

- Representasikan fakta dalam bentuk proposisi logika
S1,2: ada Stench di kotak (1,2)
B2,1: ada Breeze di kotak (2,1)
- Buat aturan dasar (pengetahuan mengenai lingkungan)
R1: $\neg S1,1 \Rightarrow \neg W1,1 \wedge \neg W1,2 \wedge \neg W2,1$
- Buat penerjemah (Translator) pengetahuan menjadi aksi
T1: $A2,1 \wedge TimurA \wedge P3,1 \Rightarrow \neg Maju$
- Temukan pengetahuan baru dan lakukan aksi yang bersesuaian, dan seterusnya sampai ditemukan solusi.

Aturan (*Rule*) Inferensi

- R1: $\neg S_{1,1} \Rightarrow \neg W_{1,1} \wedge \neg W_{1,2} \wedge \neg W_{2,1}$
- R2: $\neg S_{2,1} \Rightarrow \neg W_{1,1} \wedge \neg W_{2,1} \wedge \neg W_{2,2} \wedge \neg W_{3,1}$
- R3: $\neg S_{1,2} \Rightarrow \neg W_{1,1} \wedge \neg W_{1,2} \wedge \neg W_{2,2} \wedge \neg W_{1,3}$
- R4: $S_{1,1} \Rightarrow W_{1,3} \vee W_{1,2} \vee W_{2,2} \vee W_{1,1}$
- ...
- R33: $\neg B_{1,1} \Rightarrow \neg P_{1,1} \wedge \neg P_{1,2} \wedge \neg P_{2,1}$
- R34: $B_{2,1} \Rightarrow P_{1,1} \vee P_{2,1} \vee P_{2,2} \vee P_{3,1}$
- ...

Langkah 01

- Saat agen di kotak (1,1)
- Percept: [None, None, None, None, None]
- Modus ponens untuk $\neg S_{1,1}$ dan R1, diperoleh
 $\neg W_{1,1} \wedge \neg W_{1,2} \wedge \neg W_{2,1}$
- Lakukan Eliminasi AND, diperoleh
 $\neg W_{1,1} \quad \neg W_{1,2} \quad \neg W_{2,1}$
- Lakukan Modus Ponens, terhadap $\neg B_{1,1}$ dan R33, diperoleh
 $\neg P_{1,1} \wedge \neg P_{1,2} \wedge \neg P_{2,1}$
- Lakukan Eliminasi AND, diperoleh
 $\neg P_{1,1} \quad \neg P_{1,2} \quad \neg P_{2,1}$
- Dihasilkan 6 kalimat baru, saat agen di kotak (1,1)
- Kemana agen dapat melangkah? Ke (1,2) atau (2,1)

Langkah 02

- Agen berada di (2,1)
- Apa yang diterima (*percept*) dari lingkungan?
- Lakukan inferensi...
- Dan seterusnya...

First Order Predicate Logic (FOPL)

- **Kalkulus Predikat:** Predikat digunakan untuk mengekspresikan properti suatu obyek
- Pada **kalkulus proposisi**, “Saya suka coklat” dapat diwakili P dan $\neg P$ berarti “saya tidak suka coklat”
- Pada **logika predikat**, ditulis: **Suka(Saya, Cokelat)**, dimana Suka adalah predikat. Hubungan antara Saya dan Cokelat juga terlihat jelas.

Quantifier

- Kalimat “Saya suka coklat” dapat diekspan menjadi “ semua orang suka coklat”, berbentuk:

$$\forall x \text{ Orang}(x) \Rightarrow \text{Suka}(x, \text{Cokelat})$$

- \forall berarti “untuk semua”, disebut *universal quantifier*.
- Juga terdapat *existential quantifier* yang memperlihatkan bahwa hanya beberapa (minimal satu) yang punya properti tertentu, tidak semua:

$$\exists x \text{ Suka}(x, \text{Cokelat})$$

- Dapat dibaca “ada suatu x sedemikian hingga x suka coklat”.
- Catatan:

$$\forall x \text{ Suka}(x, \text{cokelat}) \Rightarrow \exists x \text{ Suka}(x, \text{cokelat}) \text{ adalah benar;}$$

$$\exists x \text{ Suka}(x, \text{cokelat}) \Rightarrow \forall x \text{ Suka}(x, \text{cokelat}) \text{ adalah salah.}$$

Inferensi pada FODL

- 7 pada logika Proposisi + 3 aturam tambahan

- *Universal Elimination:*

$$\forall v \alpha \Rightarrow \text{SUBSTS}(\{v/g\}, \alpha)$$

$\forall x \text{ Suka}(x, \text{Cokelat})$, x dapat digantikan oleh Saya, sehingga dapat disimpulkan $\text{Suka}(\text{Saya}, \text{Cokelat})$

- *Existential Elimination:*

$$\exists v \alpha \Rightarrow \text{SUBSTS}(\{v/k\}, \alpha)$$

$\exists x \text{ Suka}(x, \text{Cokelat})$, x dapat digantikan oleh Saya, sehingga dapat disimpulkan $\text{Suka}(\text{saya}, \text{Cokelat})$

- *Existential Introduction:*

$$\alpha \Rightarrow \exists v \text{ SUBSTS}(\{g/v\}, \alpha)$$

Dari $\text{Suka}(\text{saya}, \text{Cokelat})$ dapat disimpulkan $\exists x \text{ Suka}(x, \text{Cokelat})$.

Contoh: Hukum Pernikahan

- Pernikahan tidak sah jika kedua mempelai mempunyai hubungan keponakan.
- Wati menikah dengan Andi
- Wati anak kandung Budi
- Budi Saudara Kembar Andi
- Buktikan bahwa pernikahan Andi dan Wati tidak sah!
- Langkah-langkah:
 - Buat kalimat-kalimat dalam *First Order (Predicate) Logic*, berdasarkan pengetahuan awal yang diberikan
 - Gunakan aturan inferensi untuk membuat kalimat baru sampai diperoleh kesimpulan.

1: Membangun Kalimat Awal

1: $\forall x,y$ Keponakan(x,y) \wedge Menikah (y,x)

$\Rightarrow \neg$ Sah(Menikah(y,x))

2: Menikah(Andi, Wati)

3: AnakKandung(Wati, Budi)

4: SaudaraKembar(Budi, Andi)

5: $\forall x,y$ SaudaraKembar(x,y) \Rightarrow SaudaraKandung (y,x)

6: $\forall x,y,z$ AnakKandung(x,y) \wedge SaudaraKandung (y,z)

\Rightarrow Keponakan(x,z)

2: Proses Inferensi

- (5) dan *Universal Elimination*:
7: SaudaraKembar(Budi, Andi) \Rightarrow SaudaraKandung (Budi, Andi)
- (4) , (7) dan Modus Ponens:
8: SaudaraKandung(Budi, Andi)
- (6) dan *Universal Elimination*:
9: AnakKandung(Wati, Budi) \wedge SaudaraKandung(Budi, Andi)
 \Rightarrow Keponakan(wati, Andi)
- (3), (8) dan Pemunculan AND :
10: AnakKandung(Wati, Budi) \wedge SaudaraKandung(Budi, Andi)

2: Proses Inferensi

- (9), (10) dan Modus Ponens:

11: Keponakan(Wati, Andi)

- (1) dan Universal Elimination:

12: Keponakan(Wati, Andi) \wedge Menikah (Andi,Wati)

$\Rightarrow \neg$ Sah(Menikah(Andi,Wati))

- (11), (2), Pemunculan AND:

13: Keponakan(Wati, Andi) \wedge Menikah (Andi, Wati)

- (12), (13) dan Modus Ponens:

14: \neg Sah(Menikah(Andi, Wati))

Tugas

- Pelajari dengan seksama mengenai Logika *First Order*. Carikan contoh lain yang dapat diselesaikan dengan bahasa logika ini.
- Tuliskan ekspresi ini dalam logika first order:
 - Semua mahasiswa informatika suka kecerdasan buatan
 - Setiap yang paham pemrograman suka kecerdasan buatan
 - **Karena itu**, semua mahasiswa ilmu komputer paham pemrogramanBenarkah kesimpulan di atas?
- Ubahlah kalimat berikut ke dalam logika first order:
 - Setiap apel atau pear adalah buah
 - Setiap buah punya warna kuning atau hijau atau merah
 - Tidak ada pear berwarna merah
 - Tidak ada buah yang manis berwarna hijau

Benarkah kalimat “**Jika pear tidak kuning maka pear tidak manis**” ? Buktikan!