2025

Modul 2: First-Order Logic

REPRESENTASI PENGETAHUAN & PENALARAN

ARI SAPTAWIJAYA

Modul 2

First-Order Logic

Daftar Isi

Sebelum sebuah sistem berbasis pengetahuan (knowledge-based systems) dapat bernalar, diperlukan cara untuk mengekspresikan pengetahuan. Dalam hal ini, kita membutuhkan sebuah bahasa yang deklaratif (bahwa meyakini sesuatu yang kita ketahui, misalnya P, berarti menyatakan bahwa P bernilai benar) dan presisi (ada sintaks dari kalimat yang menyatakan apa yang kita ketahui sesuai dengan semantik dari kalimat tersebut). Dalam modul ini, kita akan fokus pada First-Order Logic (FOL) sebagai bahasa representasi pengetahuan. Kita akan mempelajari sintaks dan semantik dari FOL, serta penggunaannya dalam menarik suatu kesimpulan dari pengetahuan yang sudah ada (konsekuensi logis).

Capaian Pembelajaran

1. Peserta mampu membangun formula dalam First-Order Logic (FOL) sesuai dengan sintaks FOL
2. Peserta mampu menentukan semantik dari formula FOL berdasarkan interpretasi yang diberikan
3. Peserta mampu menentukan apakah sebuah formula adalah konsekuensi logis dari suatu himpunan formula dengan menggunakan definisi entailment dalam konteks sistem berbasis pengetahuan (knowledge-based system).

Referensi

Ronald J. Brachman and Hector J. Levesque. Knowledge Representation and Reasoning. Morgan Kaufmann, 2004 (Bab 2)

2.1 Sintaks

Setelah menyelesaikan bagian ini, Anda diharapkan mampu membangun formula yang *well-formed* (sesuai dengan aturan sintaks) dalam *First-Order Logic* (FOL).

2.1.1 Alfabet: Komponen Dasar

Setiap bahasa memiliki alfabet yang merupakan komponen dasar dari sintaks. Alfabet dalam FOL terdiri dari simbol-simbol dibagi menjadi dua kategori:

- **Simbol Logis:** simbol yang yang maknanya tetap dan tidak bergantung pada domain.
  - **Tanda Baca:** `(`, `)`, `,`
  - **Konektif:** $\neg$ (negasi), $\wedge$ (konjungsi/“dan”), $\vee$ (disjungsi/“atau”), $\forall$ (“untuk semua”), $\exists$ (“terdapat”), $=$ (kesamaan).
  - **Variabel:** x, y, z, x_1, x_2, $\ldots$ (biasanya huruf kecil).
- **Simbol Non-Logis:** simbol yang maknanya bergantung pada domain yang direpresentasikan (*domain-dependent*) .
  - **Simbol Predikat:** Menunjukkan relasi atau properti. Contoh: `Person`, `OlderThan`, `Dog`. Ditulis dengan huruf kapital.
  - **Simbol Fungsi:** Menunjukkan fungsi yang memetakan objek ke objek lain. Contoh: `fatherOf`, `successor`. Ditulis dengan huruf kecil.

Setiap simbol non-logis memiliki **aritas** (*arity*), yaitu jumlah argumen yang dibutuhkannya. Secara khusus:

- **Predikat aritas 0:** Simbol proposisional (misal: `Raining`).
- **Fungsi aritas 0:** Simbol **konstanta** (misal: `johnSmith`, `a`, `b`).

2.1.2 Grammar: Aturan Membangun Formula

Dengan menggunakan alfabet, kita dapat mendefinisiakn aturan (*grammar*) untuk membangun menjadi ekspresi yang bermakna.

1. Terms

Terms adalah ekspresi yang merujuk ke **objek** dalam domain yang dapat didefinisikan secara rekursif:

1. Setiap **variabel** adalah *term*. (Contoh: x)
2. Jika f adalah simbol fungsi dengan aritas n, dan $t_1, \ldots, t_n$ adalah *terms*, maka $f(t_1, \ldots, t_n)$ adalah *term*. (Contoh: `fatherOf(johnSmith)`, di mana `johnSmith` adalah konstanta/fungsi aritas 0).

2. Formula Atomik

Formula atomik adalah formula paling dasar yang bisa bernilai benar atau salah.

1. Jika P adalah simbol predikat aritas n, dan $t_1, \ldots, t_n$ adalah *terms*, maka $P(t_1, \ldots, t_n)$ adalah formula atomik.

   (Contoh: `OlderThan(fatherOf(johnSmith), johnSmith)`).
2. Jika t_1 dan t_2 adalah *terms*, maka $(t_1 = t_2)$ adalah formula atomik. (Contoh: `fatherOf(anna) = johnSmith`).

3. Well-Formed Formulas (wffs)

Dengan menggunakan formula atomik sebagai formula paling dasar, kita dapat membangun formula FOL yang lebih kompleks menggunakan konektif dan *quantifier* secara rekursif:

1. Setiap formula atomik adalah *wff* .
2. Jika α dan β adalah *wffs* dan v adalah variabel, maka formula berikut juga merupakan *wffs* :
   - $\neg\alpha$ (Negasi)
   - $(\alpha \wedge \beta)$ (Konjungsi)

- $(\alpha \vee \beta)$ (Disjungsi)
- $\exists v.\alpha$ (Kuantifikasi Eksistensial: "terdapat v yang memenuhi α")
- $\forall v.\alpha$ (Kuantifikasi Universal: "untuk semua v berlaku α")

Untuk operator implikasi ($\supset$) dan bikondisional atau ekivalen ($\equiv$) dapat dinyatakan sebagai berikut.

Abbreviations:

$(\alpha \supset \beta)$ for $(\neg\alpha \vee \beta)$

safer to read as disjunction than as "if ... then ..."

$(\alpha \equiv \beta)$ for $((\alpha \supset \beta) \wedge (\beta \supset \alpha))$

2.1.3 Variabel dan Kalimat (Sentences)

- **Scope Variabel:** Sama seperti dalam bahasa pemrograman, *quantifier* $(\forall, \exists)$ menentukan *scope* terikatnya sebuah variabel.
  - Dalam $P(x) \wedge \exists x[Q(x)]$, x pada $P(x)$ adalah **free** (bebas), sedangkan x pada $Q(x)$ adalah **bound** (terikat) oleh $\exists x$.
- **Kalimat (Sentence):** Sebuah *wff* yang **tidak memiliki variabel bebas** disebut sebagai kalimat (atau *closed wff*). Dalam representasi pengetahuan, kita hanya menggunakan *kalimat* dalam *knowledge base* kita.

Latihan 1

Diberikan simbol:

- Predikat: `Student(x)`, `Likes(x, y)`
- Fungsi: `advisorOf(x)`
- Konstanta: `mathClass`, `ani`

Manakah dari berikut ini yang merupakan *wff* yang valid (dan mana yang merupakan *sentence*)?

1. `Likes(ani, mathClass)`
2. `Student(advisorOf(x))`
3. $\forall x(Student(x) \wedge Likes(x, mathClass))$

2.2 Semantik

Setelah menyelesaikan bagian ini, Anda diharapkan mampu menentukan semantik (makna/nilai kebenaran) dari sebuah formula FOL berdasarkan interpretasi yang diberikan.

Sintaks hanya memberi kita aturan untuk membentuk formula menggunakan simbol-simbol. Semantik menghubungkan simbol-simbol tersebut dengan "dunia" yang direpresentasikan. Kita tidak bisa tahu makna/kebenaran sebuah kalimat tanpa mengetahui bagaimana memaknai simbol-simbol non-logisnya. Dalam FOL, untuk memahami kebenaran sebuah kalimat, kita perlu mengetahui:

1. Objek apa saja yang ada di dunia.
2. Objek mana yang memenuhi predikat P.
3. Pemetaan apa yang dilambangkan oleh fungsi f.

2.2.1 Interpretasi: $\mathfrak{F} = \langle D, I \rangle$

Semantik dalam FOL didefinisikan oleh sebuah **Interpretasi** ($\mathfrak{F}$), yang terdiri dari dua bagian :

1. **Domain (D):** Himpunan tidak kosong dari **objek** yang ada di "dunia" kita. Ini bisa apa saja: orang, angka, bahkan unicorn.
2. **Mapping Interpretasi (I):** Sebuah fungsi yang memetakan simbol nonlogis ke makna sebenarnya di dalam D .
   - **Untuk Konstanta (c):** $I[c]$ adalah satu objek spesifik di dalam D.
     - Contoh: $I[ani]$ merujuk pada suatu objek $o \in D$
   - **Untuk Simbol Fungsi (f) aritas n:** $I[f]$ adalah sebuah fungsi n-ary atas D (dari $D^n \to D$).
     - Contoh: $I[fatherOf]$ adalah fungsi yang memetakan satu objek orang ke objek orang lainnya (ayahnya).
   - **Untuk Simbol Predikat (P) aritas n:** $I[P]$ adalah sebuah relasi n-ary atas D (sebuah himpunan *tuple* $\subseteq D^n$).
     - Contoh: $I[OlderThan]$ adalah himpunan semua pasangan $\langle d_1, d_2 \rangle$ di mana d_1 lebih tua dari d_2.

2.2.2 Makna (Denotasi) dari Terms

Terms (seperti `fatherOf(ani)`) merujuk ke objek spesifik di D.

- Jika *term* memiliki variabel (misal: `fatherOf(x)`), denotasinya bergantung pada nilai x. Kita menggunakan **variable assignment** (μ) yang memetakan variabel ke objek di D .

- Denotasi $||t||_{\mathfrak{F},\mu}$ (dibaca: denotasi t untuk $\mathfrak{F}$ dan μ):
  1. Jika t adalah variabel v, maka $||v||_{\mathfrak{F},\mu} = \mu(v)$.
  2. Jika t adalah $f(t_1, \ldots, t_n)$, maka $||f(t_1, \ldots, t_n)||_{\mathfrak{F},\mu} = H(d_1, \ldots, d_n)$, di mana $H = I[f]$ dan $d_i = ||t_i||_{\mathfrak{F},\mu}$.

2.2.3 Kebenaran dari Formula (Wffs)

Sekarang kita bisa menentukan apakah sebuah formula α **dipenuhi** (*satisfied*) oleh interpretasi $\mathfrak{F}$ dan assignment μ. Kita tulis ini sebagai: $\mathfrak{F}, \mu \models \alpha$, yang didefinisikan secara rekursif sebagai berikut (jhj dibaca "jika dan hanya jika":

1. **Formula Atomik (Predikat):**
   $\mathfrak{F}, \mu \models P(t_1, \ldots, t_n)$ jhj $\langle d_1, \ldots, d_n\rangle \in I[P]$, di mana $d_i = ||t_i||_{\mathfrak{F},\mu}$. *(Artinya: $P(t_1, \ldots, t_n)$ benar jika tuple denotasi dari term-nya ada di dalam himpunan relasi untuk P).*
2. **Formula Atomik (Kesamaan):**
   $\mathfrak{F}, \mu \models (t_1 = t_2)$ jhj $||t_1||_{\mathfrak{F},\mu}$ adalah objek yang sama dengan $||t_2||_{\mathfrak{F},\mu}$.
3. **Formula dengan konektif:**
   - $\mathfrak{F}, \mu \models \neg\alpha$ jhj $\mathfrak{F}, \mu \not\models \alpha$ (tidak benar bahwa α terpenuhi).
   - $\mathfrak{F}, \mu \models (\alpha \wedge \beta)$ jhj $\mathfrak{F}, \mu \models \alpha$ dan $\mathfrak{F}, \mu \models \beta$.
   - $\mathfrak{F}, \mu \models (\alpha \vee \beta)$ jhj $\mathfrak{F}, \mu \models \alpha$ atau $\mathfrak{F}, \mu \models \beta$.
4. **Formula dengan quantifier:**
   - $\mathfrak{F}, \mu \models \exists v.\alpha$ jhj **terdapat** $d \in D$ sehingga $\mathfrak{F}, \mu\{d; v\} \models \alpha$.
     ($\mu\{d; v\}$ artinya assignment μ yang sama, tapi v dipetakan ke d).
   - $\mathfrak{F}, \mu \models \forall v.\alpha$ jhj **untuk semua** $d \in D$, $\mathfrak{F}, \mu\{d; v\} \models \alpha$.

Jika α adalah *sentence* (tidak punya variabel bebas), nilainya tidak bergantung μ. Kita bisa langsung menulis $\mathfrak{F} \models \alpha$ (α benar untuk interpretasi $\mathfrak{F}$) .

Latihan 2

Diberikan interpretasi $\mathfrak{F} = \langle D, I\rangle$ di mana:

- $D = \{1, 2, 3\}$
- $I[a] = 1$, $I[b] = 2$
- $I[P] = \{1, 2\}$ (Relasi aritas 1)
- $I[Q] = \{\langle 1, 2\rangle\}$ (Relasi aritas 2)

Tentukan apakah kalimat berikut benar atau salah untuk $\mathfrak{F}$:

1. `P(a)`

2. `Q(a, b)`
3. $\exists x P(x)$
4. $\forall x P(x)$
5. $\forall x \exists y Q(x, y)$

2.3 Konsekuensi Logis dan Sistem Berbasis Pengetahuan

Setelah menyelesaikan bagian ini, Anda diharapkan mampu menentukan apakah sebuah formula adalah konsekuensi logis (*logically entailed*) dari suatu himpunan formula (*Knowledge Base*) dengan menggunakan definisi *entailment*.

2.3.1 Konsekuensi Logis (Entailment)

Ada beberapa hubungan antar kalimat yang berlaku terlepas dari interpretasi simbol non-logis yang menyusunnya. Contoh: Jika α benar, maka $\neg(\beta \wedge \neg\alpha)$ pasti benar, tidak peduli apa interpretasi α atau β. Hal ini membawa kita ke konsep **Entailment** (Konsekuensi Logis).

Definisi Entailment:

> Sebuah himpunan kalimat S **entails** sebuah kalimat α (dinotasikan dengan $S \models \alpha$) jhj untuk **setiap** interpretasi $\mathfrak{F}$, jika $\mathfrak{F} \models S$ (semua kalimat di S benar di $\mathfrak{F}$), maka $\mathfrak{F} \models \alpha$ (α juga benar di $\mathfrak{F}$) .

Dengan kata lain, α adalah **konsekuensi logis** dari S jika tidak ada interpretasi $\mathfrak{F}$ yang menyebabkan S benar namun α salah, atau:

$$S \models \alpha \text{ jhj } S \cup \{\neg\alpha\} \text{ adalah } \textit{unsatisfiable.}$$

2.3.2 Knowledge-Based Systems (KBS)

Mengapa *entailment* penting? Karena dalam dunia nyata, kita tidak memiliki akses ke "interpretasi yang dimaksud" (*intended interpretation*) oleh pengguna. Dengan entailment, kita tahu bahwa jika S benar dalam interpretasi yang dimaksud oleh pengguna, maka α juga benar.

Perhatikan, menggunakan definisi entailment, kita tidak bisa membuktikan `Dog(fido)` $\models$ `Mammal(fido)` karena mungkin saja ada interpretasi $\mathfrak{F}$ di mana $I[Dog] \not\subseteq I[Mammal]$.Ide Kunci KR: Kita harus **menyatakan hubungan kedua objek tersebut secara eksplisit** di dalam *Knowledge Base* (KB) kita:

- $S = \{\forall x[Dog(x) \supset Mammal(x)]\}$.
- Sekarang, kita bisa buktikan: $S \cup \{Dog(fido)\} \models Mammal(fido)$.

Sebuah **Knowledge Base (KB)** adalah himpunan kalimat yang kita yakini benar (pengetahuan eksplisit), sedangkan semua kalimat α di mana $KB \models \alpha$ merupakan pengetahuan implisit.

Tugas dari sistem berbasis pengetahuan adalah melakukan **penalaran (reasoning)** atau **inferensi deduktif** untuk menjawab *entailment* ini—menjadikan pengetahuan implisit menjadi eksplisit .

deductive inference:

process of calculating entailments of KB

i.e given KB and any α, determine if KB |= α

Process is sound if whenever it produces α, then KB |= α

does not allow for plausible assumptions that may be true in the intended interpretation

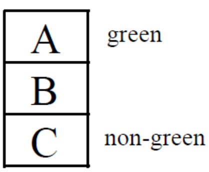
Process is complete if whenever KB |= α, it produces α

does not allow for process to miss some α or be unable to determine the status of α

2.3.3 Contoh Kasus: Tumpukan Balok

Mari kita gunakan definisi *entailment* untuk melakukan inferensi deduktif.

Problem: Ada tiga balok bertumpuk (A di atas B, B di atas C). Balok paling atas (A) berwarna hijau. Balok paling bawah (C) tidak hijau. **Pertanyaan:** Apakah ada balok hijau yang berada tepat di atas balok non-hijau?



1. **Formalisasi (KB):**
   - $S = \{On(a,b), On(b,c), Green(a), \neg Green(c)\}$
2. **Kueri (Query α):**
   - $\alpha = \exists x \exists y [Green(x) \wedge \neg Green(y) \wedge On(x,y)]$
3. **Pembuktian (Apakah $S \models \alpha$?)**
   Kita harus menunjukkan bahwa **setiap** interpretasi $\mathfrak{F}$ yang membuat S benar, pasti juga membuat α benar .

   Mari kita ambil $\mathfrak{F}$ sembarang yang memenuhi S ($\mathfrak{F} \models S$). Karena $\mathfrak{F} \models S$, kita tahu:

3.1. $\mathfrak{F} \models On(a, b)$

3.2. $\mathfrak{F} \models On(b, c)$

3.3. $\mathfrak{F} \models Green(a)$

3.4. $\mathfrak{F} \models \neg Green(c)$

Sekarang, kita periksa balok b. Di $\mathfrak{F}$ ini, hanya ada dua kemungkinan untuk b :

- **Kasus 1:** $\mathfrak{F} \models Green(b)$ **(Balok b hijau).**
  - Dari (4), kita tahu $\mathfrak{F} \models \neg Green(c)$.
  - Dari (2), kita tahu $\mathfrak{F} \models On(b, c)$.
  - Maka, $\mathfrak{F} \models Green(b) \wedge \neg Green(c) \wedge On(b, c)$.
  - Jika $x = b$ dan $y = c$, maka $\mathfrak{F} \models \exists x \exists y[Green(x) \wedge \neg Green(y) \wedge On(x, y)]$.
  - Jadi, $\mathfrak{F} \models \alpha$.
- **Kasus 2:** $\mathfrak{F} \not\models Green(b)$ **(Balok b tidak hijau).**
  - Ini berarti $\mathfrak{F} \models \neg Green(b)$.
  - Dari (3), kita tahu $\mathfrak{F} \models Green(a)$.
  - Dari (1), kita tahu $\mathfrak{F} \models On(a, b)$.
  - Maka, $\mathfrak{F} \models Green(a) \wedge \neg Green(b) \wedge On(a, b)$.
  - Jika $x = a$ dan $y = b$, maka $\mathfrak{F} \models \exists x \exists y[Green(x) \wedge \neg Green(y) \wedge On(x, y)]$.
  - Jadi, $\mathfrak{F} \models \alpha$.

Kesimpulan:
Dalam kedua kasus yang mungkin terjadi, jika $\mathfrak{F} \models S$, maka $\mathfrak{F} \models \alpha$. Oleh karena itu, terbukti bahwa $S \models \alpha$.

2.4 Latihan Modul

1. Diberikan simbol fungsi dan *arity*-nya $a/0$, $f/1$, dan $g/2$ dan dua intepretasi $\mathcal{I}_1$ dan $\mathcal{I}_2$:

   - $\mathcal{I}_1 = \langle D, I \rangle$ dengan $D = \mathbb{N}$, $I[a] = 0$, $I[f] = suc$, dan $I[g] = add$, masing-masing didefinisikan sebagai $suc(x) = x + 1$, dan $add(x, y) = x + y$;
   - $\mathcal{I}_2 = \langle D, I \rangle$ dengan D adalah himpunan *string* yang dibentuk dari alfabet {o,n}, $I[a] = \mathrm{o}$, $I[f] = add_{\mathrm{n}}$, dan $I[g] = conc$, masing-masing didefinisikan sebagai $add_{\mathrm{n}}(x) = x * \mathrm{n}$ dan $conc(x, y) = x * y$. Notasi $*$ merupakan operasi *string concatenation*.

   Berikan denotasi dari kedua term berikut dengan mengisi tabel:

Interpretasi ($\mathcal{I}$)	μ	$\|\|g(f(a), f(f(a)))\|\|_{\mathcal{I}}$	$\|\|g(f(x), g(f(a), f(x)))\|\|_{\mathcal{I},\mu}$
$\mathcal{I}_1$	$[x/4]$		
$\mathcal{I}_2$	$[x/\text{ono}]$		

2. Diberikan simbol fungsi dan *arity*-nya $a/0$ dan $s/1$, simbol predikat dan *arity*-nya $Nat/1$, dan interpretasi $\mathcal{I} = \langle D, I \rangle$ dengan $D = \{\bigcirc, \triangle, \square\}$, $I[a] = \triangle$, $I[s] : f$, didefinisikan sebagai $f(\triangle) = \square$, $f(\square) = \triangle$, $f(\bigcirc) = \bigcirc$, dan $I[Nat] = \{\triangle, \square\}$;

   Untuk kedua formula/kalimat berikut:

   - $F : \forall x[a \neq s(x)]$, dengan $t_1 \neq t_2$ didefinisikan sebagai $\neg(t_1 = t_2)$
   - $G : Nat(a) \wedge \forall x[Nat(x) \supset Nat(s(x))]$

   tunjukkan apakah $\mathcal{I} \models F$, apakah $\mathcal{I} \models G$, dan apakah $\mathcal{I} \models \{F, G\}$