FAKULTAS
ILMU
KOMPUTER

2025

Modul 4: Resolusi

REPRESENTASI PENGETAHUAN & PENALARAN

ARI SAPTAWIJAYA

Modul 4

Resolusi

Daftar Isi

Fokus modul ini adalah pada aturan inferensi **Resolusi**, sebuah prosedur pembuktian yang fundamental dalam sistem berbasis pengetahuan. Modul ini akan membahas tiga pilar utama: (1) Sifat *sound* dan *refutation-complete* dari resolusi, (2) Proses konversi formula First-Order Logic (FOL) ke Conjunctive Normal Form (CNF), dan (3) Penerapan resolusi untuk membuktikan *entailment* dalam suatu knowledge-based system.

Capaian Pembelajaran

1. Mahasiswa mampu menjelaskan resolusi sebagai aturan inferensi yang sound dan refutation-complete.

2. Mahasiswa mampu mengubah formula First-Order Logic ke bentuk yang ekivalen dalam Conjunctive Normal Form.

3. Mahasiswa mampu menerapkan aturan inferensi resolusi untuk menjawab entailment dalam knowledge-based systems yang berbasiskan first-order logic.

Referensi

Ronald J. Brachman and Hector J. Levesque. Knowledge Representation and Reasoning. Morgan Kaufmann, 2004 (Bab 4)

4.1 Resolusi sebagai Aturan Inferensi

Dalam bagian ini kita akan membahas resolusi sebagai aturan inferensi yang bersifat *sound* dan *refutation-complete*.

4.1.1 Latar Belakang: Menunjukkan Entailment

Dalam sistem berbasis pengetahuan, kita sering ingin menentukan apakah sebuah fakta α secara logis di-entail (disimpulkan) oleh suatu knowledge base (KB). Kita menuliskannya sebagai:

$$KB \models \alpha$$

Untuk membuktikannya, kita akan menggunakan pendekatan yang disebut **pembuktian dengan refutasi** (proof by refutation) atau *proof by contradiction*. Idenya adalah sebagai berikut:

$$KB \models \alpha \quad \text{jika dan hanya jika} \quad (KB \land \neg\alpha) \text{ tidak satisfiable (unsatisfiable)}$$

Sebuah formula dikatakan *unsatisfiable* jika formula tersebut bernilai *false* untuk semua interpretasi. Jika kita dapat menunjukkan bahwa $KB \cup \{\neg\alpha\}$ unsatisfiable atau dengan kata lain menunjukkan adanya kontradiksi ($FALSE$), maka kita telah membuktikan bahwa $KB \models \alpha$.

Aturan inferensi resolusi adalah prosedur untuk menemukan kontradiksi tersebut.

4.1.2 Aturan Resolusi (Kasus Proposisional)

Resolusi diterapkan pada formula dalam bentuk **Conjunctive Normal Form (CNF)**, yaitu formula yang dinyatakan sebagai konjungsi (AND) dari klausa-klausa, di mana setiap klausa adalah disjungsi (OR) dari literal-literal.

Clausal representation

Formula = set of clauses

Clause = set of literals

Literal = atomic sentence or its negation

positive literal and negative literal

Notation:

If ρ is a literal, then $\bar{\rho}$ is its complement

$\overline{p} \Rightarrow \neg p \qquad \overline{\neg p} \Rightarrow p$

To distinguish clauses from formulas:

[and] for clauses: $[p, \bar{r}, s]$ { and } for formulas: $\{ [p, \bar{r}, s], [p, r, s], [\,\overline{p}\,] \}$

[] is the empty clause {} is the empty formula

So {} is different from {[]}!

Interpretation:

Formula understood as <u>conjunction</u> of clauses	$\{[p,\neg q], [r], [s]\}$	[ ]
Clause understood as <u>disjunction</u> of literals	represents	represents
Literals understood normally	$((p \vee \neg q) \wedge r \wedge s)$	FALSE

Perhatikan, bahwa setiap formula proposisi dapat diubah menjadi formula ekivalen dalam CNF dengan tahapan-tahapan berikut.

Every propositional wff α can be converted into a formula α' in <u>C</u>onjunctive <u>N</u>ormal <u>F</u>orm (CNF) in such a way that $\models \alpha \equiv \alpha'$.

1. eliminate $\supset$ and $\equiv$ using $(\alpha \supset \beta) \twoheadrightarrow (\neg\alpha \vee \beta)$ etc.
2. push $\neg$ inward using $\neg(\alpha \wedge \beta) \twoheadrightarrow (\neg\alpha \vee \neg\beta)$ etc.
3. distribute $\vee$ over $\wedge$ using $((\alpha \wedge \beta) \vee \gamma) \twoheadrightarrow ((\alpha \vee \gamma) \wedge (\beta \vee \gamma))$
4. collect terms using $(\alpha \vee \alpha) \twoheadrightarrow \alpha$ etc.

Latihan 1

Ubahlah formula-formula berikut menjadi CNF:

- $q \supset p \vee r$
- $r \vee p \supset s$

Aturan Inferensi Resolusi: Diberikan dua klausa yang mengandung literal yang saling komplemen (misalnya p dan $\neg p$):

- Klausa 1: $\{p\} \cup C_1$

- Klausa 2: $\{\neg p\} \cup C_2$

Dengan aturan inferensi resolusi, kita dapat menurunkan klausa baru yang disebut **resolvent**:

$$C_1 \cup C_2$$

Kasus Spesial: Jika kita meresolusi $[p]$ dan $[\neg p]$, resolvent-nya adalah C_1 dan C_2 sama-sama kosong, sehingga kita mendapatkan **klausa kosong** ([ ]). Klausa kosong merepresentasikan kontradiksi atau $FALSE$.

Jika diberikan himpunan S yang terdiri dari klausa, maka kita dapat mendefinisikan *derivation* suatu klausa c dari S (dituliskan $S \rightarrow c$) sebagai berikut.

A <u>derivation</u> of a clause c from a set S of clauses is a sequence $c_1, c_2, ..., c_n$ of clauses, where $c_n = c$, and for each c_i, either

1. $c_i \in S$, or
2. c_i is a resolvent of two earlier clauses in the derivation

Latihan 2

Jika diberikan himpunan S yang berisi kedua klausa yang diperoleh dari Latihan 1 sebelumnya bersama-sama dengan klausa $[\neg p, s]$, tunjukkan bahwa klausa $[\neg q, s]$ dapat diperoleh dari S.

Akhirnya, dengan menggunakan derivation kita dapat menunjukkan apakah suatu formula dapat di-entail oleh sekumpulan formula menggunakan pendekatan pembuktian dengan refutasi yang disampaikan sebelumnya.

So: given a finite KB, to find out if KB $\models \alpha$, it will be sufficient to

1. put (KB $\wedge \neg\alpha$) into CNF, as above
2. determine the satisfiability of the clauses

Dalam hal ini, kita menunjukkan unsatisfiability dengan memperoleh derivation klausa kosong dari $(KB \wedge \alpha)$.

So for any set S of clauses: S is unsatisfiable iff $S \rightarrow []$.

Provides method for determining satisfiability: search all derivations for [].
So provides a method for determining all entailments

Latihan 3

Jika KB berisi formula-formula berikut (sebagian dari Latihan 1 dan 2 di atas) :

- $q \supset p \vee r$
- $r \vee p \supset s$
- $p \supset s$

- q

Tunjukkan dengan menggunakan derivasi pada resolusi bahwa $KB \models s$.

4.1.3 Soundness

Resolusi adalah aturan inferensi yang *sound*. Artinya, jika kita menurunkan sebuah klausa c dari himpunan klausa S ($S \rightarrow c$), maka c di-entail secara logis oleh S ($S \models c$).

Untuk menunjukkan sifat soundness dari resolusi, kita perlu membuktikan terlebih dahulu bahwa resolvent yang dihasilkan tahapan resolusi di-entail oleh klausa input-nya.

Bukti: Misalkan kita punya interpretasi $\mathfrak{I}$ yang membuat kedua klausa input bernilai *true*.

$$\mathfrak{I} \models (p \vee C_1) \quad \text{dan} \quad \mathfrak{I} \models (\neg p \vee C_2)$$

Ada dua kemungkinan untuk p:

- **Kasus 1: $\mathfrak{I} \models p$.** Karena $\mathfrak{I} \models (\neg p \vee C_2)$, dan $\mathfrak{I}$ membuat $\neg p$ *false*, maka $\mathfrak{I}$ harus membuat C_2 *true*.

- **Kasus 2: $\mathfrak{I} \not\models p$.** Karena $\mathfrak{I} \models (p \vee C_1)$, dan $\mathfrak{I}$ membuat p *false*, maka $\mathfrak{I}$ harus membuat C_1 *true*.

Dalam kedua kasus, salah satu dari C_1 atau C_2 harus *true*. Oleh karena itu, disjungsi-nya, $(C_1 \vee C_2)$, juga pasti *true*. Jadi, $\mathfrak{I} \models (C_1 \vee C_2)$. Ini membuktikan bahwa resolvent $(C_1 \cup C_2)$ adalah konsekuensi logis dari klausa input.

Selanjutnya, pembuktian di atas dapat dikembangkan untuk menunjukkan soundness dari resolusi dengan menggunakan induksi terhadap panjang dari derivation.

Can extend the previous argument to derivations:

If $S \rightarrow c$ then $S \models c$

Proof: by induction on the length of the derivation.
Show (by looking at the two cases) that $S \models c_i$.

4.1.4 Refutation Completeness

Completeness berarti jika $S \models c$, maka $S \rightarrow c$. Resolusi **tidak** *complete* secara umum.

- **Contoh:** $\{\neg p\} \models (\neg p \vee \neg q)$, tetapi kita tidak bisa menurunkan $[\neg p, \neg q]$ dari $\{\neg p\}$ menggunakan resolusi.

Namun, resolusi memiliki properti yang disebut **Refutation Completeness**. Artinya: Suatu himpunan klausa S adalah *unsatisfiable* ($S \models []$) **jika dan hanya jika** klausa kosong ($[]$) dapat diturunkan dari S menggunakan resolusi ($S \rightarrow []$).

Properti inilah yang membuat resolusi menjadi prosedur pembuktian yang dapat diandalkan untuk menjawab entailment yang yang dilakukan dengan cara **menemukan kontradiksi**.

4.2 Konversi Formula FOL ke CNF

Resolusi mengharuskan semua formula dalam KB dan negasi dari query memiliki bentuk **Conjunctive Normal Form (CNF)**. Untuk formula dalam First-Order Logic (FOL), proses konversinya memiliki tahapan tambahan dari proses konversi pada logika proposisional karena adanya *quantifier* ($\forall, \exists$) dan variabel.

Berikut adalah langkah-langkah untuk mengubah formula FOL α menjadi himpunan klausa (CNF):

1. **Eliminasi Implikasi dan Biimplikasi ($\supset, \equiv$).** Gunakan ekuivalensi standar:
   - $(\alpha \supset \beta) \equiv (\neg\alpha \vee \beta)$
   - $(\alpha \equiv \beta) \equiv (\alpha \supset \beta) \wedge (\beta \supset \alpha)$
2. **Geser Negasi ke Dalam ($\neg$).** Gunakan hukum De Morgan dan ekuivalensi *quantifier* untuk menggeser $\neg$ sehingga hanya berlaku pada atom (literal):
   - $\neg(\alpha \wedge \beta) \equiv (\neg\alpha \vee \neg\beta)$
   - $\neg(\alpha \vee \beta) \equiv (\neg\alpha \wedge \neg\beta)$
   - $\neg(\neg\alpha) \equiv \alpha$
   - $\neg\forall x.\alpha \equiv \exists x.\neg\alpha$
   - $\neg\exists x.\alpha \equiv \forall x.\neg\alpha$
3. **Standardisasi Variabel.** Pastikan setiap *quantifier* menggunakan nama variabel yang unik.
   - **Contoh:** $\exists x[P(x)] \wedge Q(x)$ menjadi $\exists z[P(z)] \wedge Q(x)$
4. **Eliminasi Quantifier Eksistensial ($\exists$) (Skolemization).** Ini adalah langkah paling krusial. Kita mengganti variabel yang terikat oleh kuantor $\exists$ dengan sebuah "nama" (konstanta atau fungsi Skolem).
   - Jika $\exists y$ tidak berada dalam cakupan $\forall$ (universal quantifier), ganti y dengan **konstanta Skolem** baru (nama yang belum pernah dipakai), misal a.
     - **Contoh:** $\exists x.P(x)$ menjadi $P(a)$.
   - Jika $\exists y$ berada dalam cakupan $\forall x_1, ..., \forall x_n$, ganti y dengan **fungsi Skolem** baru $f(x_1, ..., x_n)$.
     - **Contoh:** $\forall y \exists x.R(x, y)$ menjadi $\forall y.R(f(y), y)$.

Catatan Penting: Skolemization *tidak* menjamin sifat ekivalensi, tetapi menjamin sifat *satisfiability*, yang cukup untuk kebutuhan pembuktian refutasi.

5. **Pindahkan Quantifier Universal ($\forall$) ke Depan.** Karena semua $\exists$ sudah hilang, kita dapat memindahkan semua $\forall$ ke awal formula tanpa mengubah makna (disebut sebagai Bentuk Normal Prenex).
   - **Contoh:** $(\forall x.\alpha) \wedge \beta \equiv \forall x(\alpha \wedge \beta)$ (jika β tidak mengandung x).
6. **Distribusi $\vee$ terhadap $\wedge$.** Gunakan hukum distributif untuk mengubah formula menjadi CNF.
   - $((\alpha \wedge \beta) \vee \gamma) \equiv ((\alpha \vee \gamma) \wedge (\beta \vee \gamma))$

   Hasilnya adalah konjungsi dari disjungsi literal.
7. **Sederhanakan menjadi Klausa.**
   - **Buang kuantor $\forall$:** Karena semua variabel yang tersisa terikat secara universal, kita bisa membuang $\forall$: semua variabel secara implisit dikuantifikasi secara universal.
   - Kumpulkan Setiap bagian dari konjungsi ($\wedge$) menjadi klausa terpisah.

     Contoh: $(P(x) \vee \neg Q(y)) \wedge (R(y) \vee S(z))$ menjadi dua klausa:

     (a) $[P(x), \neg Q(y)]$

     (b) $[R(y), S(z)]$

Latihan 4

Ubahlah formula berikut ke dalam CNF:

$$\forall x[(\forall y[P(y) \supset Q(x, y)]) \supset (\exists z\ R(z, x))]$$

4.3 Resolusi dan Entailment

Pada bagian ini kita akan menerapkan aturan inferensi resolusi untuk menjawab *entailment* dalam *knowledge-based systems.*

4.3.1 Prosedur Umum Pembuktian Entailment

Untuk membuktikan $KB \models \alpha$, kita mengikuti prosedur pembuktian dengan refutasi:

1. **Persiapan:**
   - Ambil semua formula dalam KB.

- Ambil negasi dari query, yaitu $\neg\alpha$.

2. **Konversi:** Ubah semua formula dari langkah 1 menjadi CNF (himpunan klausa S) menggunakan langkah-langkah konversi ke CNF yang telah dijelaskan sebelumnya.

3. **Resolusi:** Terapkan aturan resolusi FOL secara berulang pada pasangan klausa di S untuk menghasilkan resolvent baru. Tambahkan resolvent ini ke S.

4. **Terminasi:**
   - **SUKSES:** Jika **klausa kosong ([])** berhasil diturunkan, S adalah *unsatisfiable*. Ini membuktikan bahwa memang benar $KB \models \alpha$.
   - **GAGAL:** Jika tidak ada resolusi baru yang dapat dibuat dan [] belum ditemukan, S adalah *satisfiable*. Ini membuktikan bahwa $KB \not\models \alpha$ (Catatan: untuk FOL, proses ini tidak dijamin *terminate*).

4.3.2 Unification dalam Resolusi pada FOL

Resolusi pada FOL dapat melibatkan literal yang mengandung variabel, misal $P(x)$ dan $\neg P(a)$. Untuk menerapkan resolusi pada pasangan literal ini, kita memerlukan **Unification**.

Unification yang diterapkan pada dua buah literal adalah proses menemukan substitusi θ (sebuah *unifier*) untuk variabel-variabel sehingga dua literal tersebut dapat dibuat identik.

- $\theta = \{v_1/t_1, ..., v_n/t_n\}$ adalah substitusi di mana v_i adalah variabel dan t_i adalah term.

Dalam melakukan unifikasi pada resolusi FOL, kita mencari Most General Unifier (MGU).

Aturan Resolusi FOL (dengan MGU): Diberikan dua klausa (yang variabelnya sudah distandardisasi agar unik) yang akan diterapkan resolusi pada pasangan literal $\{\rho_1\}$ dan $\{\rho_2\}$:

- Klausa 1: $\{\rho_1\} \cup C_1$
- Klausa 2: $\{\rho_2\} \cup C_2$

Jika ρ_1 dan $\overline{\rho_2}$ (komplemen dari ρ_2) dapat di-unifikasi oleh sebuah **Most General Unifier (MGU)** θ, maka kita dapat memperoleh resolvent baru:

$$(C_1 \cup C_2)\theta$$

Misalkan, $\rho_1 = P(x, a)$ dan $\rho_2 = \neg P(b, y)$, maka $\overline{\rho_2} = P(b, y))$

- Unifikasi akan menghasilkan MGU $\theta = \{x/b, y/a\}$ sedemikian hingga $\rho_1\theta = P(b, a)$ dan $\overline{\rho_2}\theta = P(b, a)$; oleh karenanya $\rho_1\theta = \overline{\rho_2}\theta$

- Dengan MGU θ di atas, diperoleh resolvent

$$(C_1 \cup C_2)\theta$$

.

4.3.3 Menjawab Entailment dengan Resolusi

Mari kita buktikan $KB \models \text{HardWorker(sue)}$ dari contoh berikut.

$$\text{KB} \overset{?}{\models} \text{HardWorker(sue)}$$

KB

$\forall x$ GradStudent(x) $\supset$ Student(x)
$\forall x$ Student(x) $\supset$ HardWorker(x)
GradStudent(sue)

1. Konversi KB dan Negasi Query $\neg Q$ (dalam CNF):

- KB:
  - $\forall x.(\text{GradStudent}(x) \supset \text{Student}(x))$
    menjadi $[\neg\text{GradStudent}(x), \text{Student}(x)]$, kita sebut KB1
  - $\forall x.(\text{Student}(x) \supset \text{HardWorker}(x))$
    menjadi $[\neg\text{Student}(x), \text{HardWorker}(x)]$, kita sebut KB2
  - GradStudent(sue) menjadi [GradStudent(sue)], kita sebut KB3
- Q: HardWorker(sue)
- ¬Q: ¬HardWorker(sue) menjadi [¬HardWorker(sue)]

2. Proses Resolusi:

1. **Resolve** KB2 dan ¬Q:
   - $[\neg\text{Student}(x), \text{HardWorker}(x)]$
   - [¬HardWorker(sue)]
   - Literal: HardWorker(x) dan ¬HardWorker(sue)
   - MGU: $\theta = \{x/\text{sue}\}$
   - **Resolvent 1:** [¬Student(sue)]
2. **Resolve** Resolvent 1 dan KB1:

- [$\neg$Student(sue)]
- [$\neg$GradStudent(x), Student(x)]
- Literal: $\neg$Student(sue) dan Student(x)
- MGU: $\theta = \{x/\text{sue}\}$
- **Resolvent 2:** [$\neg$GradStudent(sue)]

3. **Resolve** Resolvent 2 dan KB3:
   - [$\neg$GradStudent(sue)]
   - [GradStudent(sue)]
   - Literal: $\neg$GradStudent(sue) dan GradStudent(sue)
   - MGU: $\theta = \{\}$ (kosong)
   - **Resolvent 3:** [] (Klausa Kosong)

3. Kesimpulan: Klausa kosong ([]) berhasil diperoleh. Oleh karena itu, kita telah membuktikan bahwa memang benar $KB \models$ HardWorker(sue).

4.4 Latihan Modul

1. Diberikan KB sebagai berikut:
   - $\forall x.(Rajin(x) \wedge Cermat(x) \supset Lulus(x))$
   - $\forall x.(\neg Rajin(x) \supset Malas(x))$
   - $\forall x.(\neg Cermat(x) \supset Ceroboh(x))$
   - $\forall x.[Bahagia(x) \supset (\neg Malas(x) \wedge \neg Ceroboh(x))]$
   - $\neg Lulus(budi)$

   Gunakan resolusi untuk menjawab apakah $KB \models \neg Bahagia(budi)$.

2. Di sebuah desa terpencil terdapat aturan yang berlaku untuk pemangkas rambut di desa tersebut:
   - R_1: Setiap penduduk desa yang tidak memangkas rambutnya sendiri dipangkas oleh si pemangkas rambut.
   - R_2: Setiap penduduk desa yang dipangkas oleh si pemangkas rambut tidak memangkas rambutnya sendiri.

   Jawablah pertanyaan-pertanyaan berikut:

   (a) Formulasikan aturan tersebut ke dalam kalimat FOL.

(b) Tunjukkan bahwa tidak ada pemangkas rambut di desa tersebut yang dapat memenuhi aturan yang diberikan dengan menunjukkan bahwa intepretasi apapun yang memenuhi aturan yang satu membuat aturan kedua tidak terpenuhi.

(c) Pertanyaan (b) menunjukkan bahwa $R_1 \wedge R_2$ *unsatisfiable*, yaitu $\{R_1, R_2\} \models []$. Apakah dengan resolusi juga bisa ditunjukkan bahwa $\{R_1, R_2\} \vdash []$?