שימוש בלוגיקה לייצוג ידע

תחשיב הפרדיקטים

- תחשיב הפרדיקטים זוהי שפה לייצוג טיעונים.
- לטיעונים בתחשיב הפרדיקטים יש שני מרכיבים: עצמים ותכונות שמייחסים להם. התכונות נקראות פרדיקטים (יחסים).
- לדוגמא:
  - כל הלימונים צהובים.
  - אורי סטודנט.

תחשיב הפרדיקטים (תחביר)

השפה מכילה:

1. סימני קבועים $c_1, c_2, c_3, \ldots$
2. סימני משתנים $x_1, x_2, x_3, \ldots$
3. לכל n טבעי קבוצה של סימני פונקציות n-מקומיות: $f_1^n, f_2^n, f_3^n, \ldots$
4. לכל n טבעי חיובי קבוצה של סימני פרדיקטים n-מקומיים: $p_1^n, p_2^n, p_3^n, \ldots$
5. קשרים לוגיים: $\neg, \rightarrow, \wedge, \vee$.
6. כמתים: $\exists, \forall$

תחשיב הפרדיקטים (תחביר) - המשך

נגדיר את אוסף שמות העצם (ביטויים) בשפה:

- קבועים.
- משתנים.
- אם $t_1, t_2, \ldots, t_n$ שמות עצם ו- f_k^n סימן פונקציה n מקומית, אזי $f_k^n(t_1, t_2, \ldots, t_n)$ הינו שם עצם.

נגדיר את אוסף הנוסאות:

- אם $t_1, t_2, \ldots, t_n$ הם שמות עצם, ו- p_k^n הוא סימן פרדיקט n מקומי, אזי $p_k^n(t_1, t_2, \ldots, t_n)$ נוסחא.
  נוסחא כנ"ל נקראת פסוק אטומי.
- תהי α נוסחא, אזי $\neg\alpha$ נוסחא.
- יהיו α, β נוסחאות, אזי $\alpha \vee \beta, \alpha \wedge \beta, \alpha \to \beta$ נוסחאות.
- תהי α נוסחא ויהי x משתנה, אזי $\exists x : \alpha, \forall x : \alpha$ נוסחאות.

הצבה

הגדרות:

- משתנה קשור (bound) - משתנה המופיע בנוסחא עם כמת.
  דוגמא: $\forall x : Apple(x) \rightarrow Red(x)$.
- משתנה חופשי (free) - משתנה שאינו קשור.
  דוגמא: $Apple(x) \rightarrow Red(x)$.
- נוסחא תקרא פסוק אם לא מופיעים בה משתנים חופשיים.
- יהיו s ו-t שמות עצם, יהי x משתנה, נגדיר את $s(x|t)$ כשם העצם המתקבל מ-s כאשר כל מופע של המשתנה x מוחלף ב- t.
- תהי α נוסחא ו-t שם עצם, יהי x משתנה, נגדיר את $\alpha(x|t)$ כנוסחא המתקבלת מ-α כאשר כל מופע חופשי של המשתנה x מוחלף בשם העצם t.

האקסיומות הלוגיות

קבוצת האקסיומות הלוגיות היא קבוצת הנוסחאות הבאה:

1. $\alpha \to (\beta \to \alpha)$
2. $(\alpha \to (\beta \to \gamma)) \to ((\alpha \to \beta) \to (\alpha \to \gamma))$
3. $(\neg\alpha \to \beta) \to ((\neg\alpha \to \neg\beta) \to \alpha)$
4. $(\forall x : (\alpha \to \beta)) \to (\forall x : \alpha \to \forall x : \beta)$
5. $\alpha \to \forall x : \alpha$ (כאשר x אינו חופשי ב- α)
6. $\forall x : \alpha \to \alpha(x|t)$ (כאשר t הינו שם עצם ללא משתנים חופשיים שהופכים להיות קשורים כאשר מציבים אותו במקום x)

האקסיומות הנ"ל הינן סכימות של נוסחאות. ניתן להציב כל נוסחא במקום α, β, γ לעיל כדי לקבל נוסחת אקסיומה ספציפית.

הוכחה

- תהי Φ קבוצת נוסחאות ותהי α נוסחא. הוכחה **של** α **מתוך** Φ היא סדרה סופית ולא ריקה

$$\psi_1, \psi_2, \ldots, \psi_n$$

של נוסחאות המקיימת את שני התנאים הבאים:

  - לכל $k = 1, \ldots, n$:
    - ψ_k אקסיומה לוגית; או
    - ψ_k נוסחא ב- Φ; או
    - קיימים $i, j < k$ כך ש- $\psi_i = \psi_j \to \psi_k$.
  - $\psi_n = \alpha$

כלל היסק ויכיחות

- כלל היסק - כלל ניתוק הרישא Modes ponens:
  בהינתן נוסחאות $\alpha \rightarrow \beta$ ו- α, ניתן להסיק β.
- בהינתן $\Phi = \{\alpha, \alpha \rightarrow \beta\}$ נאמר כי β יכיח מתוך Φ ונסמן $\Phi \vdash \beta$.

סמנטיקה

- תהי U קבוצת עצמים (עולם).
- פירוש I הוא פונקציה הממפה איברים של השפה לאיברים של הייצוג באופן הבא:
  1. הקבועים ממופים לאובייקטים בעולם.
  2. הפונקציות ממופות לפונקציות המתאימות בין n-יות של איברים בעולם לאיברים בעולם.
  3. הפרדיקטים ממופים ליחסים בעולם.
- השמה לכל המשתנים היא מיפוי של המשתנים החופשיים לאיברים ב-U.

ערך האמת של נוסחאות

- פונקצית הערכה של נוסחאות ממפה נוסחא ל-TRUE או FALSE:
- ערך האמת V של נוסחא α תחת פירוש I מוגדר באופן רקורסיבי:
  - אם $\alpha = P(t_1, t_2, \ldots, t_n)$ נוסחא אטומית אזי:
    - אם $I(P)\left\langle I(t_1), I(t_2), \ldots, I(t_n)\right\rangle$, אזי $V(\alpha) = \textbf{TRUE}$.
- אחרת, $V(\alpha) = \textbf{FALSE}$.
- אם $\alpha = \neg\beta$ אזי $V(\alpha)$ יקבע עפ"י טבלת האמת של $\neg$ עבור $V(\beta)$.
- אם $\alpha = \beta \vee \gamma$ או $\alpha = \beta \wedge \gamma$ או $\alpha = \beta \rightarrow \gamma$ אזי $V(\alpha)$ יקבע עפ"י טבלת האמת של הקשרים $\vee, \wedge, \rightarrow$ בהתאמה עבור $V(\beta)$ ו- $V(\gamma)$.
- אם $\alpha = \forall x : \beta$ אזי:
  - אם לכל השמה של איבר מ-U ב-x נקבל $V(\beta) = \textbf{TRUE}$, אז $V(\alpha) = \textbf{TRUE}$.
  - אחרת, $V(\alpha) = \textbf{FALSE}$.
- הערה: $\exists x : \beta$ שקול ל: $\neg\forall x : \neg\beta$.

טבלאות האמת

α	β	$\alpha \vee \beta$
T	**T**	**T**
T	**F**	**T**
F	**T**	**T**
F	**F**	**F**

α	$\neg\alpha$
T	**F**
F	**T**

α	β	$\alpha \to \beta$
T	**T**	**T**
T	**F**	**F**
F	**T**	**T**
F	**F**	**T**

α	β	$\alpha \wedge \beta$
T	**T**	**T**
T	**F**	**F**
F	**T**	**F**
F	**F**	**F**

ספיקות

- פירוש I (interpretation) והשמה V מספקים נוסחא אם"ם הנוסחא מקבלת ערך TRUE תחת הפירוש וההשמה.
- קבוצת פסוקים Φ גוררת לוגית **פסוק** α אם"ם כל פירוש והשמה שמספקים את Φ מספקים גם את α. נסמן: $\Phi \models \alpha$.
- משפט השלמות: נוסחא נובעת לוגית מקבוצת נוסחאות אם"ם היא יכיחה מתוכן:

$$\Phi \vdash \alpha \Leftrightarrow \Phi \models \alpha$$

- משפט השלמות מקשר בין סינטקס לסמנטיקה.
- משפט השלמות מספק כלי לבדיקת גרירה לוגית.

ייצוג ידע (Knowledge Representation)

- ייצוג עצמים:

Table (my_desk)

Car (my_suzuki)

- ייצוג תכונות של עצמים:

Num_of_legs(my_desk,4)

Color (my_suzuki, silver)

- ייצוג תכונות כלליות של עצמים:

$\forall x : [\text{Table}(x) \rightarrow \text{Num\_ of\_ legs}(x, 4)]$

- ייצוג יחסים בין עצמים:

Parent (avraham, izak)

On (coffee_glass, my_desk)

ייצוג ידע - המשך

- ייצוג יחסים כלליים:

הגדרת יחס Above לפי קואורדינאטת גובה של עצמים:

$$\forall x \forall y [\mathrm{Greater}(\mathrm{z\_coord}(x), \mathrm{z\_coord}(y)) \rightarrow \mathrm{Above}(x, y)]$$

- ייצוג תכונות של יחסים:

טרנזיטיביות של יחס Above:

$$\forall x \forall y \forall z [\mathrm{Above}(x, y) \wedge \mathrm{Above}(y, z) \rightarrow \mathrm{Above}(x, z)]$$

- ייצוג היררכיות:

$$\forall x : [\mathrm{Work\_Table}(x) \rightarrow \mathrm{Table}(x)]$$

$$\forall x : [\mathrm{Table}(x) \rightarrow \mathrm{Num\_of\_legs}(x, 4)]$$

$$\mathrm{Work\_Table}(\mathrm{my\_desk})$$

ייצוג ידע - דוגמא

המירו את המשפטים הבאים לייצוג לוגי:

- Jack owns a dog.
- Every dog owner is an animal lover.
- No animal lover kills an animal.
- Either Jack or Curiosity killed the cat, who is named Tuna.
- Did Curiosity kill the cat?

פרוצדורת הוכחה

- פרוצדורה שבהינתן משפט (נוסחא יכיחה) תמצא לו הוכחה.
- אם נוסחא אינה משפט אזי לא מובטח לנו שפרוצדורה כזו תעצור!

הפרוצדורה:

1. $S \leftarrow \emptyset$.
2. הוסף אקסיומה ל- S לפי מניה כלשהי של האקסיומות.
3. הפעל את כללי ההיסק על נוסחאות ב-S.
4. הוסף את הנוסחאות החדשות ל-S.
5. אם הנוסחא המבוקשת נמצאת ב-S, החזר "כן".
6. אחרת, חזור ל-2.

רזולוציה

- פרוצדורה יעילה יותר ופשוטה יותר למימוש הינה הרזולוציה.
- פרוצדורה זו מסתמכת על מבנה קנוני של נוסחאות CNF (Conjuctive Normal Form):

$$(L_{11} \vee L_{12} \vee \cdots \vee L_{1n^1}) \wedge \cdots \wedge (L_{m1} \vee L_{m2} \vee \cdots \vee L_{mn^m})$$

- L_{ij} נקרא ליטרל - פסוק אטומי או שלילת פסוק אטומי.
- כל אחת מהשורות נקראת פסוקית (clause).
- לנוסחת CNF מתייחסים כקבוצת פסוקיות ולכל פסוקית כקבוצת ליטרלים.

אלגוריתם הרזולוציה (ללא משתנים)

1. הנחה: האקסיומות והנוסחא אותה נרצה להוכיח חסרות משתנים.
2. קלט: קבוצת אקסיומות A, נוסחא P (אותה רוצים להוכיח)
3. הפוך את $A \cup \{\neg P\}$ לקבוצת פסוקיות E.
4. אתחל $D \leftarrow E$.
5. בחר שתי פסוקיות ב- D מהצורה:

$$A = \{A_1, \ldots A_n, L\}; B = \{B_1, \ldots B_m, \neg L\}$$

6. עדכן: $D \leftarrow D \cup \{\{A_1, \ldots A_n, B_1, \ldots B_m\}\}$.
7. אם לא ניתן להמשיך - החזר "לא".
8. אם נמצאה סתירה ($\emptyset \in D$) - החזר "כן".
9. אם משאבי החישוב שהוקצו נצרכו - החזר "לא ידוע".
10. חזור ל- 5.

אלגוריתם הרזולוציה (ללא משתנים) - הערות

- כדי לקבל פסוקית ריקה, D צריך להכיל שתי פסוקיות $\{L\}$ ו-$\{\neg L\}$. זוג פסוקיות כאלו מובילות לסתירה כיוון שבין כל הפסוקיות קיים קשר $\wedge$.
- פרוצדורת הרזולוציה מבצעת הוכחה בדרך השלילה (מוכיחה כי שלילת המשפט מובילה לסתירה).
- כלל הרזולוציה שומר על נכונות.

טענות לגבי תהליך הרזולוציה

- תהליך הרזולוציה נאות (sound). כל פסוקית שנגזרה מקבוצת פסוקיות D ע"י תהליך הרזולוציה, נובעת לוגית מ-D.
- תהליך הרזולוציה הינו *refutation complete*. אם קבוצת פסוקיות D אינה ספיקה, ניתן לגזור מ-D את הפסוקית הריקה.

אלגוריתם הרזולוציה עם משתנים

- כעת נסיר את ההנחה כי האקסיומות והנוסחא אותה נרצה להוכיח חסרות משתנים.
- במקרה זה בצעד 5 של האלגוריתם, במקום לחפש שני ליטרלים זהים כאשר אחד מהם מופיע עם סימן שלילה, אנו נחפש שני ליטרלים שניתן להפוך אותם לזהים ע"י הצבה מתאימה.
- לתהליך המוצא הצבה שהופכת שני פסוקים לזהים קוראים האחדה (unification).

פרוצדורות בתהליך הרזולוציה

- כדי לממש את הרזולוציה עם משתנים עלינו להוסיף את הפרוצדורות הבאות:
  - הפיכת נוסחא כלשהי לקבוצת פסוקיות (CNF).
  - מציאת הצבה מאחדת (יוניפיקציה).

העברת נוסחאות לצורת CNF

- הסר את סימני הגרירה $\rightarrow$. השתמש בשקילות:

$$A \rightarrow B \equiv \neg A \vee B$$

- הקטן את טווח השליליות לפסוקים אטומיים. השתמש בשקילויות:

$$\begin{aligned}
\neg(\neg A) &\equiv B \\
\neg(A \wedge B) &\equiv \neg A \vee \neg B \\
\neg(A \vee B) &\equiv \neg A \wedge \neg B \\
\neg\exists x : P(x) &\equiv \forall x : \neg P(x) \\
\neg\forall x : P(x) &\equiv \exists x : \neg P(x)
\end{aligned}$$

- שנה את שמות המשתנים כך שלא יופיע אותו שם בשני כמתים.
- העבר את כל הכמתים לתחילת הנוסחא תוך שמירה על סדר הופעתם.

העברת נוסחאות לצורת CNF - המשך

- הסר את הכמתים הישיים ($\exists$):
  - יהי n מספר הכמתים הכוללים משמאל לכמת הישי. הנוסחא עם הכמת הישי נמצאת בטווח n הכמתים. יהיו $x_1, \ldots, x_n$ המשתנים של הכמתים הנ"ל.
  - החלף את כל ההופעות של המשתנה של הכמת הישי בפונקציה חדשה בעלת שם חדש כלשהו (שאינו מופיע במקום אחר) בעלת n ארגומנטים $x_1, \ldots, x_n$. הפונקציה נקראת פונקצית סקולם (Skolem function).
  - אם לא קיימים כמתים כוללים משמאל, הפונקציה תהיה בעלת 0 ארגומנטים. במקרה זה הפונקציה נקראת קבוע סקולם.

העברת נוסחאות לצורת CNF - המשך

- הסר את הכמתים הכוללים.

עכשיו קיימים רק כמתים כוללים. מסירים אותם וזוכרים שכל המשתנים הינם של כמתים כוללים.

- הפוך את הנוסחא לקוניונקציה של דיסיונקטים. השתמש בשקילות:

$$(A \wedge B) \vee C \quad \equiv \quad (A \vee C) \wedge (B \vee C)$$

- קרא לכל דיסיוניקציה clause. שנה שמות משתנים כך שבכל clause יהיו שמות אחרים. הסתמך על השקילות:

$$\forall x : [P(x) \wedge Q(x)) \equiv \forall x : (P(x)) \wedge \forall x : (Q(x))$$

מציאת הצבה מאחדת

- הצבה - אוסף של זוגות $\{x_1|t_1,\ldots x_n|t_n\}$ כאשר t_i הם ביטויים ו- x_i הם משתנים ומתקיימים התנאים הבאים:
  - $\forall i \neq j : x_i \neq x_j$
  - x_i אינו מופיע באף אחד מהביטויים $t_i,\ldots,t_n$.
- ניתן להפעיל הצבה σ על נוסחא α ולקבל נוסחא חדשה ע"י החלפת כל המשתנים בנוסחא המופיעים בהצבה בביטויים המתאימים. נסמן את הביטוי החדש ב- $\alpha\sigma$.

מציאת הצבה מאחדת - המשך

- תהי σ הצבה. אם בהצבה τ לא מופיעים משתנים המקבלים ערך בהצבה σ ניתן להגדיר את ההרכבה של τ על σ (נסמן $\sigma\tau$).
- תהי $\sigma = \{x_1|t_1, \ldots, x_n|t_n\}$. תהי τ הצבה כנ"ל. אזי:

$$\sigma\tau = \{x_1|t_1\tau, \ldots, x_n|t_n\tau\} \cup \tau$$

(כלומר מפעילים את τ על הביטויים של σ ומוסיפים את הזוגות החדשים לזוגות של τ).

- האחדה של שני ביטויים היא הצבה ההופכת אותם לזהים.

אלגוריתם ההאחדה של רובינזון

האלגוריתם מוצא האחדה σ בהניתן שני ליטרלים אם קיימת כזו.

קלט: שני ליטרלים L_1, L_2.

1. אתחל $L_0^1 \leftarrow L_1; L_0^2 \leftarrow L_2; k \leftarrow 0; \sigma_0 \leftarrow \emptyset$.
2. אם $L_k^1 = L_k^2$, סיים והחזר את σ_k.
3. תהי D_k נקודת אי-הסכמה בין L_k^1 ל- L_k^2. נסמן $D_k = \{T_j, v_j\}$,
כלומר באחד הביטויים L_k^1, L_k^2 מופיע T_j ובאחר v_j.
4. אם v_j משתנה, T_j ביטוי ו- $v_j \notin T_j$, אזי בצע:
   (א) $L_{k+1}^1 \leftarrow L_k^1\{T_j|v_j\}$.
   (ב) $L_{k+1}^2 \leftarrow L_k^2\{T_j|v_j\}$.
   (ג) $\sigma_{k+1} \leftarrow \sigma_k\{T_j|v_j\}$.
   (ד) $k \leftarrow k+1$.
   (ה) חזור ל- 2.
5. אחרת, סיים והחזר כישלון.

רזולוציה עם משתנים

נתון אוסף הנחות (נוסחאות) $A_1, \dots, A_n$ ומשפט T אותו נרצה להוכיח.

- הפוך את הנוסחא $A_1 \wedge \cdots \wedge A_n \wedge \neg T$ לאוסף של פסוקיות $\{C_1, \dots, C_m\}$.
- התחל עם אוסף פסוקיות $P = \{C_1, \dots, C_m\}$.
- בצע את הלולאה הבאה:
  - בחר 2 פסוקיות $C_h = \{L_1, \dots, L_h\},\ C_k = \{D_1, \dots, D_k\}$ כך שקיימים $D_j = \alpha, L_j = \neg\beta$ כאשר α, β ניתנים להאחדה עם הצבה σ.
  - תהי $C_{new} = [C_h \cup C_k \setminus \{L_i, D_j\}]$.
  - החלף את שמות המשתנים ב- C_{new} לשמות חדשים שלא מופיעים ב- P.
  - בצע $P \leftarrow P \cup \{C_{new}\}$.

רזולוציה עם משתנים - המשך

- תנאי העצירה:
  - נמצאה סתירה - החזר "המשפט הוכח"; או
  - לא ניתן להמשיך - החזר "המשפט לא ניתן להוכחה"; או
  - המשאבים שהוקצו נצרכו - החזר "לא נמצאה הוכחה במשאבים המוקצים".

ייצוג ידע - דוגמא

המירו את המשפטים הבאים לייצוג לוגי:

- Jack owns a dog.
- Every dog owner is an animal lover.
- No animal lover kills an animal.
- Either Jack or Curiosity killed the cat, who is named Tuna.
- Did Curiosity kill the cat?

ייצוג ידע - פתרון דוגמא

- Jack owns a dog.

$$\exists x : [\mathrm{Dog}(x) \wedge \mathrm{Own}(\mathrm{jack}, x)]$$

- Every dog owner is an animal lover.

$$\forall x : [(\exists y : \mathrm{Dog}(y) \wedge \mathrm{Own}(x, y)) \rightarrow \mathrm{Animal\_\ lover}(x)]$$

- No animal lover kills an animal.

$$\neg\exists x : [\mathrm{Animal\_\ lover}(x) \wedge \exists y : (\mathrm{Animal}(y) \wedge \mathrm{Kill}(x, y))]$$

- Either Jack or Curiosity killed the cat, who is named Tuna.

$$\mathrm{Cat}(\mathrm{tuna}) \wedge (\mathrm{Kill}(\mathrm{jack}, \mathrm{tuna}) \vee \mathrm{Kill}(\mathrm{curiosity}, \mathrm{tuna}))$$

- (A cat is an animal)

$$\forall x : [\mathrm{Cat}(x) \rightarrow \mathrm{Animal}(x)]$$

העברה ל- CNF

- $\mathrm{Dog}(\mathrm{D})$
- $\mathrm{Own}(\mathrm{jack}, \mathrm{D})$
- $\neg\mathrm{Dog}(y) \vee \neg\mathrm{Own}(x, y) \vee \mathrm{Animal\_\ lover}(x)$
- $\neg\mathrm{Animal\_\ lover}(z) \vee \neg\mathrm{Animal}(t) \vee \neg\mathrm{Kill}(z, t)$
- $\mathrm{Cat}(\mathrm{tuna})$
- $\mathrm{Kill}(\mathrm{jack}, \mathrm{tuna}) \vee \mathrm{Kill}(\mathrm{curiosity}, \mathrm{tuna}))$
- $\neg\mathrm{Cat}(w) \vee \mathrm{Animal}(w)$
- הנחת השלילה: $\neg\mathrm{Kill}(\mathrm{curiosity}, \mathrm{tuna})$

אסטרטגית רזולוציה

- נסתכל על תהליך הרזולוציה כעל חיפוש:
- מצב - הוא אוסף פסוקיות.
- מצב התחלתי - אוסף האקסיומות + שלילת המשפט בצורת CNF.
- מצב סופי - כל קבוצת פסוקיות המכילה את הפסוקית הריקה.
- פונקצית מעבר - כלל הרזולוציה: בהינתן מצב S קבוצת המצבים הבאה היא
- $\{S \cup \{r_i\} |$ רזולוציה ע"י S מתקבלת מתוך $r_i\}$