Departamento Inteligencia Artificial

LÓGICA - 1º Grado en Ingeniería Informática

Facultad de Informática
Universidad Politécnica de Madrid

Lógica Computacional: Estandarización de Fórmulas

Petr Sosík
psosik@fi.upm.es
http://web3.fi.upm.es/AulaVirtual/
Despacho 2201

Estandarización de Fórmulas

- Objetivo: Simplificar las fórmulas para técnicas de deducción automáticas (computables)
  - Queremos obtener, mediante una serie de transformaciones, una fórmula que sea más fácil de manipular automáticamente, pero que siga teniendo ciertas propiedades de la fórmula original (estandarización)

- *Fórmula en un lenguaje de primer orden*
  - $\exists y\ (\ \forall x\ (P(x,f(y)) \rightarrow Q(z)) \vee \neg \exists w\ P(g(w),y))$

- *Fórmula en forma normal de Skolem*
  - $\forall x \forall w (\neg P(x,f(b)) \vee Q(a) \vee \neg P(g(w),b))$

- *Fórmula en forma clausular*
  - $\{\neg P(x,f(b)) \vee Q(a) \vee \neg P(g(w),b)\}$

Estandarización de Fórmulas

- Forma Normal de Skolem (FNS):
  - Todos los cuantificadores a la cabeza de la fórmula (forma Prenex)
  - No hay variables libres
  - Sólo hay cuantificadores universales
  - Matriz de la fórmula (subfórmula tras los cuantificadores) en forma normal conjuntiva (FNC, conjunción de disyunciones de literales)

- Método para obtener la FNS de cualquier fórmula:
  1. Poner la fórmula en forma Prenex
  2. Realizar el cierre existencial
  3. Poner la fórmula en forma normal conjuntiva
  4. Eliminar los cuantificadores existenciales

Forma Prenex

- Se utilizarán los siguientes teoremas de equivalencia:
  - Cambio de nombre de variables ligadas:
    - $\vdash \forall x A(x) \leftrightarrow \forall y A(x/y)$ $\quad$ $\vdash \exists x A(x) \leftrightarrow \exists y A(x/y)$ $\quad$ si y no está ya en A
    - Si x es ligada por un $\forall$ o $\exists$ y en el mismo tiempo aparece en la formula también libre o ligada por otro cuantificador, ¡debe ser renombrada! $\quad$ $\forall x A(x) \lor \forall x C(x) \leftrightarrow \forall x \forall y\, (A(x) \lor C(y))$
  - Interdefinición de cuantificadores:
    - $\vdash \neg\forall x A(x) \leftrightarrow \exists x \neg A(x)$ $\quad$ $\vdash \neg\exists x A(x) \leftrightarrow \forall x \neg A(x)$
  - Distribución de conectivas respecto a cuantificadores
    - $\vdash \forall x A \land C \leftrightarrow \forall x(A \land C)$ $\quad$ $\vdash (\forall x A \to C) \leftrightarrow \exists x(A \to C)$
    - $\vdash \exists x A \land C \leftrightarrow \exists x(A \land C)$ $\quad$ $\vdash (\exists x A \to C) \leftrightarrow \forall x(A \to C)$
    - $\vdash \forall x A \lor C \leftrightarrow \forall x(A \lor C)$ $\quad$ $\vdash (C \to \forall x A) \leftrightarrow \forall x(C \to A)$
    - $\vdash \exists x A \lor C \leftrightarrow \exists x(A \lor C)$ $\quad$ $\vdash (C \to \exists x A) \leftrightarrow \exists x(C \to A)$

    Si x está libre en la otra subfórmula, ¡debe ser renombrado!

- Lema: Para toda la fórmula A, $\vdash A \leftrightarrow Prenex(A)$
- Lema: La forma Prenex de una fórmula siempre existe, aunque puede no ser única

Departamento Inteligencia Artificial

Cierre Existencial

- Las variables libres de la fórmula se ligan existencialmente poniendo el cuantificador correspondiente en cabeza de la fórmula

- Lema: Una fórmula $A(x)$ es satisfacible sii $\exists x A(x)$ es satisfacible

- Ejemplo:
  - $\forall y \exists z (p(x) \wedge q(y) \rightarrow r(f(z), x))$ se transformaría en $\exists x \forall y \exists z (p(x) \wedge q(y) \rightarrow r(f(z), x))$

Departamento Inteligencia Artificial

Forma Normal Conjuntiva (FNC)

- Se utilizarán los siguientes teoremas de equivalencia:
  - Interdefinición:
    - $\vdash (A \rightarrow B) \leftrightarrow (\neg A \vee B)$
      Si A o B contiene una conjunción, debe cerrar este A o B a paréntesis.
      P.ej. $\vdash (p(x) \rightarrow q(x) \wedge r(x)) \leftrightarrow (\neg\, p(x) \vee (q(x) \wedge r(x)))$
    - $\vdash (A \leftrightarrow B) \leftrightarrow (A \rightarrow B) \wedge (B \rightarrow A)$
  - Leyes de Morgan
    - $\vdash \neg(A \wedge B) \leftrightarrow \neg A \vee \neg B$
    - $\vdash \neg(A \vee B) \leftrightarrow \neg A \wedge \neg B$
  - Distribución de $\vee$ y $\wedge$
    - $\vdash (A \wedge B) \vee (A \wedge C) \leftrightarrow A \wedge (B \vee C)$
    - $\vdash A \vee (B \wedge C) \leftrightarrow (A \vee B) \wedge (A \vee C)$
- Lema: Para toda fórmula A, $\vdash A \leftrightarrow \mathrm{FNC}(A)$
- Lema: La forma normal conjuntiva (FNC) de una fórmula siempre existe

Departamento Inteligencia Artificial

Eliminación de cuantificadores existenciales (Skolemización)

- Se elimina el cuantificador existencial sustituyendo la variable que ligaba por una función de Skolem o constante de Skolem

- La función de Skolem será una función <u>nueva</u> en la fórmula, aplicada a todas la variables cuantificadas universalmente que aparecen <u>antes</u> que el cuantificador existencial a eliminar. Si no hay tales variables se utilizará una <u>constante</u> nueva en la fórmula.

- Ejemplos:
  - $\forall x \exists y (P(x) \rightarrow \neg Q(y))$ se transformaría en $\forall x (P(x) \rightarrow \neg Q(f(x))$
  - $\exists x \forall z (Q(x,z) \vee R(a,x))$ se transformaría en $\forall z (Q(b,z) \vee R(a,b))$
  - $\exists x \forall y \exists z (P(x) \wedge Q(y) \rightarrow R(f(z), x))$ se transformaría en $\forall y (P(a) \wedge Q(y) \rightarrow R(f(g(y)), a))$

- Lema: Una fórmula A es satisfacible sii Skolem(A) es satisfacible

Que preserva la transformación a FNS?

- Consideremos una formula A transformada a su FNS A’
  - La transformación preserva la satisfacibilidad: A es satisfacible sii A’ es satisfacible (aunque pueden tener modelos distintos)
  - Pero NO preserva todos los modelos: no es verdad que, para toda interpretación i, i es un modelo de A sii es un modelo de A’
  - Entonces el resultado **NO** es equivalente a la fórmula original

Departamento Inteligencia Artificial

Analysis detallada de la preservación

- **Teorema: Una fórmula A es satisfacible sii FNS(A) es satisfacible**
  1. A satisfacible sii Prenex(A) satisfacible (por lemas 1 y 5)
  2. Prenex(A) satisfacible sii Cierre(Prenex(A)) satisfacible (por lema 2)
  3. Sea Cierre(Prenex(A)) satisfacible de la forma Q.M, donde Q son los cuantificadores y M es la matriz de la fórmula. Entonces M satisfacible sii FNC(M) satisfacible (por lemas 3 y 5)
  4. Pero también Q.M satisfacible sii Q. FNC(M) satisfacible, puesto que cuantifican las mismas variables en idéntica forma
  5. Q.FNC(M) satisfacible sii Skolem(Q.FNC(M)) satisfacible (por lema 4). Pero Skolem(Q.FNC(M)) es precisamente FNS(A)
  6. A satisfacible sii FNS(A) satisfacible (por silogismo 1, 2, 3, 4, 5)

  - Lema 1: Para toda fórmula A, $\vdash A \leftrightarrow$ Prenex(A)
  - Lema 2: Una fórmula A(x) es satisfacible sii $\exists x A(x)$ es satisfacible
  - Lema 3: Para toda fórmula A, $\vdash A \leftrightarrow$ FNC(A)
  - Lema 4: Una fórmula A es satisfacible sii Skolem(A) es satisfacible.
  - Lema 5: Si $\vdash A \leftrightarrow A'$ entonces: A satisfacible sii A' satisfacible

Departamento Inteligencia Artificial

Forma Clausular

- Sabemos que “Una fórmula A es satisfacible sii FNS(A) es satisfacible” y que “FNS(A) existe siempre para cualquier fórmula A”, luego **podemos trabajar exclusivamente con fórmulas en forma normal de Skolem** si sólo tuviéramos que comprobar la satisfacibilidad

- Para trabajar más cómodamente con fórmulas en FNS utilizaremos la forma clausular
  - Cláusula: Disyunción de literales
  - La forma clausular de una fórmula A es el conjunto de cláusulas de la FNS(A)
  - La forma clausular se entiende como la conjunción de las cláusulas, cuyas variables están todas ellas ligadas universalmente.

- Ejemplo:
  - A: $\forall x \forall y(P(x) \wedge (Q(y) \vee R(a,x)))$
  - FC(A): $\{P(x), Q(y) \vee R(a,x)\}$

- Teorema: Una fórmula A es satisfacible sii FC(A) es satisfacible

Forma clausular de una deducción

- Una deducción T[P1, P2, …, Pn] |– C es correcta sii P1 ∧ P2 ∧ … ∧ Pn ∧ ¬C es insatisfacible

- Dada una deducción: T[P1, P2, …, Pn] |–C
  1. Obtener la forma clausular de cada Pi, 1 ≤i ≤n
  2. Obtener la forma clausular de ¬C
  3. Realizar la unión de todos los conjuntos de cláusulas
  4. Comprobar la satisfacibilidad

Resumen

- Los pasos para expresar una fórmula en forma clausular:
  1. Forma Prenex: preserva modelos
  2. Realizar el cierre existencial: preserva satisfacibilidad
  3. Poner la fórmula en forma normal conjuntiva: preserva modelos
  4. Eliminar los cuantificadores existenciales (forma normal Skolem): preserva satisfacibilidad
  5. Escribir la fórmula en la forma clausular (solo un cambio de notación)
- En total, transformación a la forma clausular preserva satisfacibilidad pero no modelos
- Una formula en forma clausular es satisfacible sii existe una interpretación que es modelo de todas las cláusulas
- Expresar deducción T[P1, P2, …, Pn] |– C en forma clausular:
  - Expresar las premisas P1, P2, …, Pn y negación ¬C de la conclusion en forma clausular
  - Hacer unión de todos los conjuntos de cláusulas resultantes
- Una deducción es correcta sii su forma clausular es insatisfacible

Departamento Inteligencia Artificial