Intelligenza Artificiale

Sistemi a regole
Sistemi esperti

Marco Piastra

Sistemi a regole – Sistemi esperti

1. Introduzione al calcolo dei predicati

2. Sistemi a regole

3. Jess

4. Fox, Goat and Cabbage (esercitazione Jess)

1

Introduzione al calcolo dei predicati

Limiti della logica proposizionale

- Logica proposizionale: solo *fatti*

  a: "*Socrate è un essere umano*" $\rightarrow$ b: "*Socrate è mortale*"

  a: "*Socrate è un essere umano*"

  ---

  b: "*Socrate è mortale*"

  $a \rightarrow b, a \vdash b$ (modus ponens)

- Nessuna astrazione

  u: "*Ogni essere umano*" $\rightarrow$ m: "*è mortale*"

  a: "*Socrate è un essere umano*"

  ---

  b: "*Socrate è mortale*" *

  $u \rightarrow m, a \nvdash b$ (?)

Fatti, oggetti e relazioni

- Occorre poter descrivere strutture più complesse
- I *fatti*

  "*Socrate è un essere umano*"

  "*Socrate è mortale*"

  "*Kermit è amico di Miss Piggy*"

  "*Silvia è la madre di Giorgio e di Paola*"

- sono espressi in riferimento a *oggetti*

  "*Socrate*", "*Kermit*", "*Miss Piggy*", ...

- e *relazioni* *(o concetti)*

  "*essere umano*", "*mortale*", "*amico*", "*madre*", ...

LP' – Predicati e costanti

- Linguaggio proposizionale rivisitato: $\mathscr{L}_P$'
- Al posto dell'insieme di **lettere proposizionali** (*atomiche*)
- si adottano:
  - un insieme di **simboli predicativi**, con un numero predefinito di **argomenti**

    p.es. essereUmano(.), mortale(.), amico(..), madre(..), ...
  - un insieme di **costanti individuali**

    p.es. Socrate, Kermit, MissPiggy, Silvia, Giorgio, Paola, ...
- Ogni *fatto* deve essere espresso in LP usando predicati e costanti
  - In forma atomica: un solo *simbolo predicativo* con una *costante* per ciascun *argomento*
  - Esempi di **fbf**:

    essereUmano(Socrate)

    essereUmano(Socrate) $\rightarrow$ mortale(Socrate)

    madre(Silvia, Giorgio) $\wedge$ madre(Silvia, Paola)

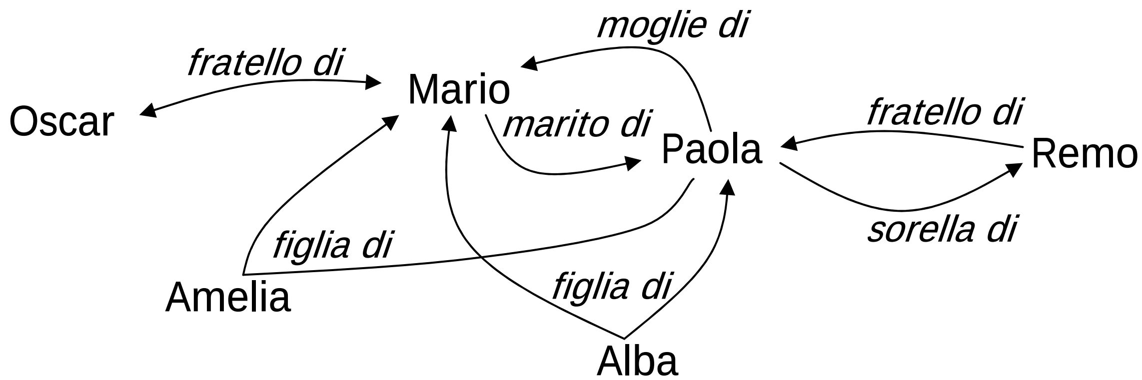
Non si può usare un *simbolo predicativo* come *argomento*

LP' – Interpretazioni simboliche

- Un'**interpretazione** di $\mathscr{L}_P$' è una funzione
  v : fbf($\mathscr{L}_P$') $\rightarrow$ {0, 1}
- Definizione alternativa v'
  - sulla base dell'insieme dei **simboli predicativi** e delle **costanti** di $\mathscr{L}_P$'
  - si costruisce un insieme H che contiene i fatti elementari che la v' rende veri (cioè a cui v' associa il valore 1)
  - Esempio:
    - $\mathscr{L}_P$' : {Socrate, Giunone}, {mortale(.), essereUmano(.)}
    - H = {mortale(Socrate), essereUmano(Socrate)}
    - v'(mortale(Socrate)) = 1 sse mortale(Socrate) $\in$ H
    - v'(mortale(Giunone)) = 0 sse mortale(Socrate) $\notin$ H
  - il valore delle fbf composite viene associato in base alle usuali regole
    - v'(mortale(Socrate) $\wedge$ mortale(Giunone)) = 1
      sse mortale(Socrate) e mortale(Giunone) $\in$ H
    - v'(mortale(Socrate) $\vee$ mortale(Giunone)) = 1
      sse mortale(Socrate) o mortale(Giunone) $\in$ H
    - v'($\neg$mortale(Giunone)) = 1 sse mortale(Giunone) $\notin$ H

LP’ – Interpretazioni, modelli

- Modello intuitivo



- **Interpretazione** formale
  - Struttura formale (fatti atomici)

    H = {fratelloDi(Oscar, Mario), fratelloDi(Mario,Oscar), moglieDi(Mario,Paola). maritoDi(Paola,Mario), figliaDi(Mario,Amelia), figliaDi(Paola,Amelia), figliaDi(Mario,Alba), figliaDi(Paola,Alba), fratelloDi(Paola,Remo), sorellaDi(Remo,Paola)}

  - fbf soddisfatte da H (cioè che H rende vere):

    maritoDi(Paola,Mario)
    fratelloDi(Oscar, Mario) $\wedge$ fratelloDi(Mario,Oscar)
    maritoDi(Paola, Remo) $\rightarrow$ (figliaDi(Mario, Amelia) $\wedge$ figliaDi(Mario, Alba))

LPred – Variabili e quantificatore

- Si arricchisce il linguaggio: $\mathscr{L}_{\mathrm{Pred}}$
- A $\mathscr{L}_{\mathrm{P}}$' si aggiungono:
  - un insieme di **variabili**
    - p.es. x, y, z, ...
  - il **quantificatore universale** $\forall$
- Regole (aggiuntive) di buona formazione:
  - è ammesso l'uso delle variabili al posto delle costanti individuali
    - p.es. mortale(x), madreDi(x,y)
  - il quantificatore universale è associato ad una variabile e si usa come un connettivo unario
    - se $\varphi \in \mathrm{fbf}(\mathscr{L}_{\mathrm{Pred}})$ allora $(\forall x\ \varphi) \in \mathrm{fbf}(\mathscr{L}_{\mathrm{Pred}})$
  - Esempi:

    mortale(x)
    ($\forall$x mortale(x))
    ($\forall$x (essereUmano(x) $\rightarrow$ mortale(x)))

LPred – Significato intuitivo

- Le fbf **aperte** (i.e. con variabili non quantificate) indicano **astrazioni**
  - Un aggettivo predicativo (p.es. 'mortale') indica una qualità o una proprietà che gli oggetti possiedono o meno
  - la fbf 'mortale(x)'
    indica l'insieme di oggetti che possiedono questa proprietà
  - analogamente, la fbf 'madreDi(x,y)'
    indica l'insieme di coppie di oggetti per cui sussiste tale relazione
- Le fbf **chiuse** (i.e. tutte le variabili sono quantificate)
  indicano **fatti** o **regole**
  - la fbf '$\forall$x (essereUmano(x) $\rightarrow$ mortale(x))'
    indica un fatto, sebbene di carattere generale

    mentre la fbf 'essereUmano(Socrate) $\rightarrow$ mortale(Socrate)'
    indica un fatto specifico
  - solo le fbf chiuse hanno un valore di verità
    le formule aperte, in certo senso, sono strutture simboliche incomplete

LPred – Interpretazioni, modelli

- Definiamo solo una regola semplificata, in relazione ad H
  - *** Adeguata <u>solo</u> per lo studio dei sistemi a regole!**
- Un'interpretazione v di $\mathscr{L}_{\text{Pred}}$
  - si basa sulla stessa struttura H usata per $\mathscr{L}_{\text{P}}$'
  - si aggiunge la regola per il quantificatore ∀
    - sia φ una fbf in cui x è l'unica variabile ed in φ non si hanno altri quantificatori
    - $v(\forall x\ \varphi) = 1$ sse tutte le sostituzioni di x con una costante generano una fbf φ' : $v(\varphi') = 1$
    - Esempio: v(∀x essereUmano(x)) = 1 sse
      essereUmano(Socrate) ∈ H
      essereUmano(Kermit) ∈ H
      essereUmano(Silvia) ∈ H
  - la regola di interpretazione è estesa ai quantificatori multipli
    - p.es. (∀x ∀y sorellaDi(x,y))
  - ed alla composizione di formule
    - p.es. ((∀x essereUmano(x)) → (∀x mortale(x)))

LPred – Derivazioni

- Valgono le regole di inferenza, di derivazione e gli assiomi di LP
- Si aggiunge una nuova regola, di *istanziazione*
  - a partire da una formula chiusa,
    contenente un numero qualsiasi di variabili quantificate
  - si può derivare una formula chiusa da cui
    - un quantificatore viene eliminato
    - sostituendo alla corrispondenti variabile una costante individuale
    - Esempi:

$$\frac{\forall x \; (\text{essereUmano}(x) \rightarrow \text{mortale}(x))}{(\text{essereUmano}(\text{Socrate}) \rightarrow \text{mortale}(\text{Socrate}))}$$

$$\frac{\forall x \; \forall y \; \text{figliaDi}(x,y)}{\forall x \; \text{figliaDi}(x,\text{Silvia})}$$

LPred – Teorie

- Si definisce

  AK: $\forall x$ (essereUmano(x) $\rightarrow$ mortale(x))

  come l'assioma di una teoria specifica K

- Per definizione la *teoria* Σ
  è composta da tutte le formule derivabili da AK
  - quindi, tutte le *istanziazioni* di AK
  - più tutte le formule derivabili in base alle regole proposizionali
  - Esempio: derivazione di "Socrate è mortale"

    essereUmano(Socrate) $\vdash$ mortale(Socrate)

    1: $\vdash$ $\forall x$ (essereUmano(x) $\rightarrow$ mortale(x)) (AK)
    2: $\vdash$ essereUmano(Socrate) $\rightarrow$ mortale(Socrate) (ist)
    3: essereUmano(Socrate) $\vdash$ mortale(Socrate) (ded)

LPred – Derivazione e conseguenza

- Si consideri ancora la teoria K, con l'unico assioma

  AK: $\forall$x (essereUmano(x) $\rightarrow$ mortale(x))

- Si consideri un $\mathscr{L}_{\text{Pred}}$ con *infinite* costanti individuali
  – Qualsiasi modello di K costruito con una struttura H è infinito
  – La relazione $\Gamma \models \varphi$ non può più essere calcolata direttamente
    - Non si può più verificare in modo diretto la derivazione precedente
  – Ma se il calcolo dei predicati è **completo**,
  si può utilizzare il calcolo simbolico al posto di quello semantico
  – Abbiamo abbandonato la logica dei fatti specifici
  e ci addentriamo in quella delle astrazioni su oggetti e relazioni.
    - Se poi fosse anche decidibile ...

2

Sistemi a regole
(*Production Systems*)

Logica e sistemi a regole

- La logica simbolica:
  - è un sistema per la rappresentazione formale del ragionamento
  - si basa su un formalismo di rappresentazione e su regole di derivazione sintattica (regole di inferenza)

- I sistemi a regole
  - adottano una forma semplificata di programmazione logica
  - sono stati concepiti per una classe di applicazioni particolari (*sistemi esperti* o *expert systems*)

Logica - Principi generali e fatti specifici

- In una rappresentazione logico-simbolica
  (p.es. clausole di Horn)
  è tipicamente possibile distinguere:
  - la rappresentazione di *principi generali*
  - la rappresentazione di *fatti specifici*
- Esempio:

  principi generali :

  $\forall x\ \forall y\ (((\text{madre}(x) = \text{madre}(y)) \wedge (\text{padre}(x) = \text{padre}(y)) \leftrightarrow \text{StessiGenitori}(x, y))$

  $\forall x\ \forall y\ ((\text{Maschio}(x) \wedge \text{StessiGenitori}(x, y)) \leftrightarrow \text{Fratello}(x, y))$

  $\forall x\ \forall y\ ((\text{Femmina}(x) \wedge \text{StessiGenitori}(x, y)) \leftrightarrow \text{Sorella}(x, y))$

  fatti specifici :

  Femmina(Amelia); Femmina(Alba); Femmina(Paola); Maschio(Mario)

  (madre(Amelia) = Paola); (madre(Alba) = Paola)

  (padre(Amelia) = Mario); (padre(Alba) = Mario)

Regole di produzione

- Un sistema a regole contiene un insieme di regole di produzione
- Ciascuna regola ha la forma
  - <condizioni> $\Rightarrow$ <azioni>
  - talvolta anche descritte come
    <LHS - Left Hand Side> $\Rightarrow$ <RHS - Right Hand Side>
  - condizioni e azioni sono in forma normale congiuntiva (FNC)

- Esempio:

Regola “Fratello”
madre(x) = madre(y)
padre(x) = padre(y)
Maschio(x)
$\Rightarrow$
Fratello(x, y)

Regola “Sorella”
madre(x) = madre(y)
padre(x) = padre(y)
Femmina(x)
$\Rightarrow$
Sorella(x, y)

Memoria di lavoro

- Un sistema a regole include anche una *memoria di lavoro* (anche *working memory* o WM)
  - la *memoria di lavoro* contiene la rappresentazione dei *fatti specifici*
  - le *regole* operano sulla *memoria di lavoro*
  - le *condizioni* sono istanziate sulla base dei *fatti specifici*
  - le *azioni* tipicamente comportano l'asserzione o la ritrattazione di *fatti specifici* (ma non solo)

Memoria di lavoro

Femmina(Amelia)
Femmina(Alba)
(madre(Amelia) = Paola)
(madre(Alba) = Paola)
(padre(Amelia) = Mario)
(padre(Alba) = Mario)

Regola "Sorella"
madre(x) = madre(y)
padre(x) = padre(y)
Femmina(x)
⇒
Sorella(x, y)

Memoria di lavoro

Femmina(Amelia)
Femmina(Alba)
(madre(Amelia) = Paola)
(madre(Alba) = Paola)
(padre(Amelia) = Mario)
(padre(Alba) = Mario)
Sorella(Alba, Amelia)

Agenda, attivazione

- In ogni istante, un sistema a regole mantiene un *agenda* che contiene le *regole istanziate*
- Il sistema sceglie le *regole istanziate* da attivare
- L'*attivazione* (*firing*) delle regole avviene in modo sequenziale

Memoria di lavoro

Femmina(Amelia)
Femmina(Alba)
(madre(Amelia) = Paola)
(madre(Alba) = Paola)
(padre(Amelia) = Mario)
(padre(Alba) = Mario)

Agenda

Regola "Sorella" (istanza 1)

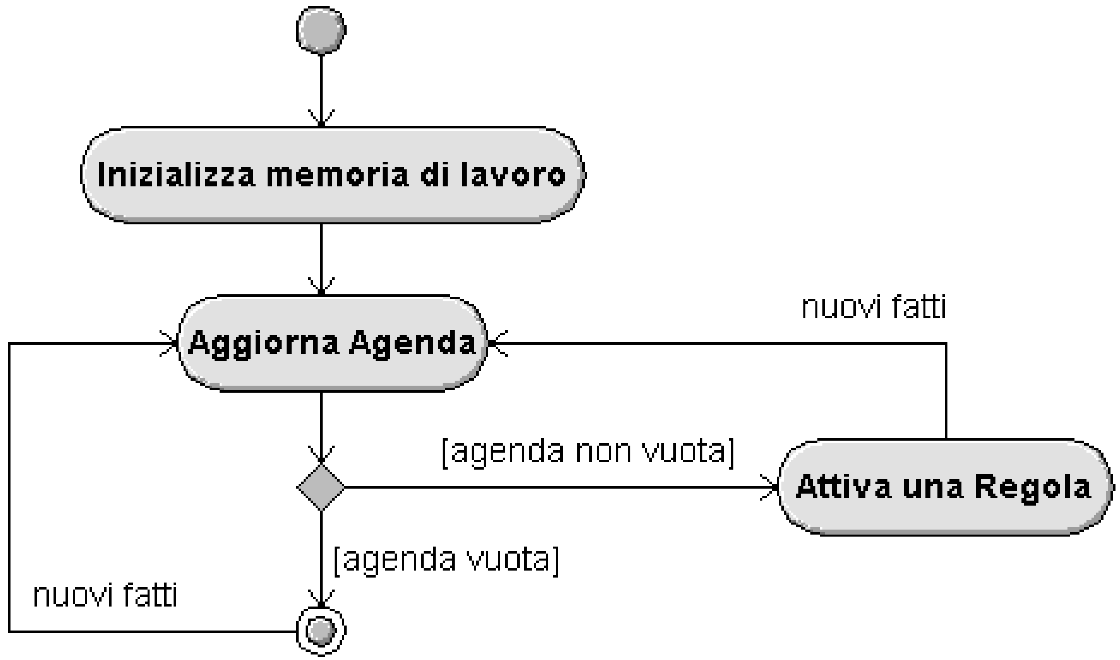
madre(Amelia) = madre(Alba)
padre(Amelia) = padre(Alba)
Femmina(Amelia)
⇒
Sorella(Amelia, Alba)

Regola "Sorella" (istanza 2)

madre(Alba) = madre(Amelia)
padre(Alba) = padre(Amelia)
Femmina(Alba)
⇒
Sorella(Alba, Amelia)

Ciclo di esecuzione

- Il sistema a regole procede ciclicamente:
  - aggiorna l'agenda
  - sceglie ed attiva una regola
  - aggiorna la memoria di lavoro

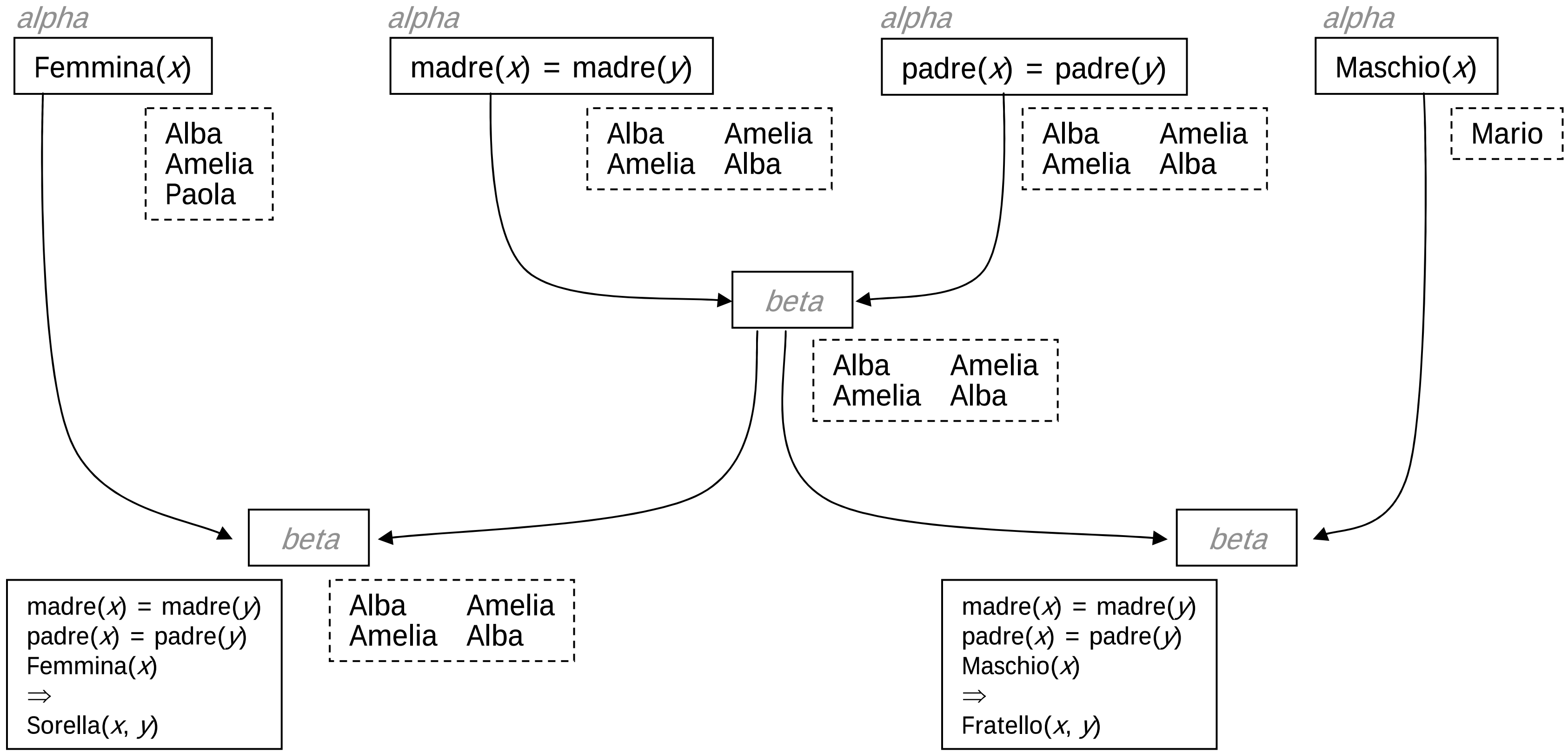


Sistema a regole - come funzionano?

- Il punto critico è l'aggiornamento dell'*agenda*
  - occorre identificare tutte le *istanziazioni* delle regole
  - evitando i cicli infiniti
    - le regole vanno inserite nell'agenda solo in presenza di *fatti nuovi*
    - diversamente, la loro attivazione è inutile
    - si veda l'esempio precedente
- Il confronto diretto è dispendioso
  - sarebbe necessario confrontare tutte le *regole* con tutti i *fatti*
    - distinguendo i fatti nuovi da quelli già noti

Algoritmo Rete (C. Forgy, 1980)

- Le *condizioni* di un *insieme di regole* vengono rappresentate in forma di *grafo aciclico*
  - a cui viene 'agganciata' la rappresentazione della *memoria di lavoro*



Aggiornamento della Rete

- I nuovi fatti vengono ‘agganciati’ ai nodi di pertinenza
  - “Gino ha gli stessi genitori di Paola”