Laurea Specialistica in Informatica
a.a. 2005-2006

Interazione Uomo-Macchina II:

Interfacce Intelligenti

Fiorella de Rosis

Programma del Corso

Introduzione

Prima parte: *Formalizzazione e Ragionamento*

- **1.1. Ragionamento logico:**
  - **Formalizzazione**
  - Risoluzione
- 1.2. Ragionamento incerto
  - Reti Causali Probabilistiche
  - Reti dinamiche
  - Apprendimento di Reti

Seconda parte: *Modelli di Utente*

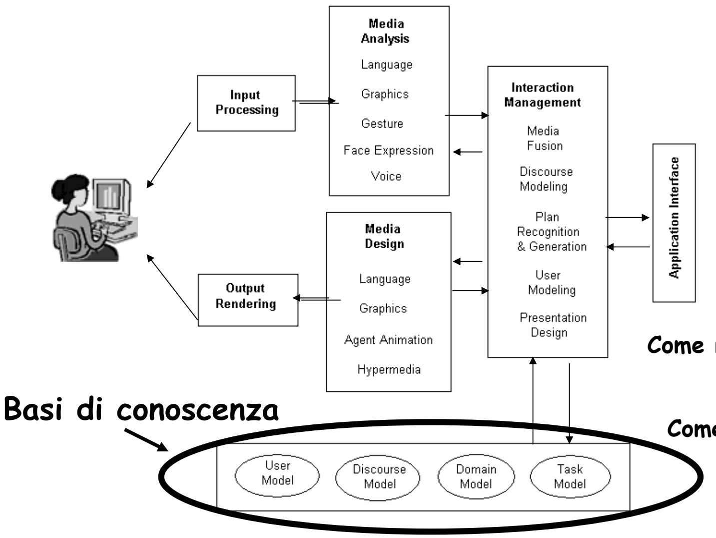
- 2.1. Modelli logici
- 2.2. Modelli con incertezza

Terza parte: *Interazione in linguaggio naturale*

- 3.1. Generazione di messaggi
  - Introduzione
  - Teorie
  - Metodi
- 3.2. Comprensione di messaggi

Quarta parte: *Simulazione di dialoghi*

Architettura di una interfaccia intelligente



Come rappresentiamo questa base di conoscenza? Come ragioniamo su questi dati?

Riprendiamo l'Es 2.4: Un dialogo di 'advice-giving'

S: Dovresti andare a correre, Giuseppe!

U: Perché?

S: Perché sei giovane, ci tieni alla tua salute, e correre fa bene alla salute.

U: Ma ti pare che, a vent'anni, già devo pensare alla salute?

S: Secondo me si. Ma comunque, correre aiuta anche a tenersi in forma.

U: E chi te l'ha detto?

S: Lo dicono studi epidemiologici svolti in diversi paesi, da istituti di ricerca qualificati.

U: Ma io detesto correre.

...

Vediamo come formalizzare la conoscenza necessaria per simulare un dialogo di questo tipo

Es 2.4:
Un esempio nel campo della persuasione

1. Task model: le strategie persuasive di Walton

a. *Appeal to positive consequences*

“Se ritieni che compiere una determinata azione comporti conseguenze importanti per te e sei in grado di compierla, dovresti farla”.

b. *Appeal to negative consequences*

“Se ritieni che compiere una determinata azione comporti conseguenze che desideri evitare e puoi evitare di compierla, dovresti farlo”.

2. Domain model

“Fare sport fa bene alla salute e alla forma fisica. Fare una vita sedentaria aumenta il rischio di ingrassare e di perdere tono muscolare”.

“Il running è un particolare tipo di sport. Tutti i giovani senza particolari problemi di salute sono in grado di fare running”.

… continuiamo l’ esempio…

3. User model

“Giuseppe è giovane,
non ha particolari problemi di salute
e tiene alla sua forma fisica.
Come molti giovani,
non si preoccupa invece, per ora, della sua salute.”

4. Discourse model

“Fare sport fa bene alla salute e alla forma fisica”, *oppure*
“Fare un po’ di sport ti fa sentire bene e in forma”... ecc
(stesso contenuto, diversa ‘realizzazione linguistica’, cioè diverso stile).

Problema:
Come faccio a convincere Giuseppe ad andare a correre?
Quali argomenti sono convincenti per lui?
Come formulo questo argomento? (con quale stile)?

Ho bisogno di un metodo per

- *Formalizzare* discorsi in linguaggio naturale e
- *Ragionare* sulla ‘conoscenza’ così espressa per derivare nuovi fatti (i quesiti a cui intendo rispondere).

Il ‘calcolo degli enunciati, in genere, non basta:
non permette di unificare ‘conoscenza generale’ e ‘conoscenza specifica’.

Linguaggi Logici Per la Formalizzazione di Discorsi in Linguaggio Naturale

Utilizzeremo una *teoria del prim’ordine per formalizzare* discorsi in linguaggio naturale.

Ad esempio:

il contenuto di un modello di utente,

le strategie di generazione di discorsi in linguaggio naturale, ...

e per ragionare su queste ‘basi di conoscenza’

Elementi del linguaggio

Termini:

Variabili (denotate con caratteri minuscoli): x, y, z, ...

Costanti (denotate con caratteri maiuscoli): A, B, C, ...

Funtori (denotati con stringhe del tipo F°) applicati a uno o più termini:

Father°(y) per 'il padre di y', Father°(MARIA)

Formule atomiche:

Predicati n-ari (denotati con stringhe che iniziano con una maiuscola) applicati a n *termini*, che hanno valore T/F:

Father(x,y) per 'x è il padre di y'; Father(PIETRO,MARIA)

Connettivi: $\wedge, \vee, \neg, \rightarrow$

Quantificatori: $\forall$, $\exists$

Formule:

combinazioni appropriate di formule atomiche con connettivi e quantificatori

Cerchiamo gli errori

Quali di queste formule non sono valide?

1. $\forall x\ (P(x) \rightarrow Q(x))$
2. $\forall x\ \exists y\ (R(x, y) \rightarrow S(x, F°(y)))$
3. $\neg\forall x\ \exists y\ (R(x, y) \rightarrow S(x, F(x,y)))$
4. $\exists y\ \forall x\ (R(x, y) \rightarrow S(x, y))$
5. $\exists x\ R(x, y)$
6. $\forall x\ \neg\exists x\ R(x, y)$

Cerchiamo gli errori

Quali di queste formule sono vere
(nella vostra interpretazione del mondo)?

1. $\forall x\ (Donna(x) \rightarrow Bella(x))$
2. $\forall x\ \exists y\ (Amico(x, y) \wedge Litiga(x, y))$
3. $\forall x\ \exists y\ (Amico(x, y) \wedge Litiga(x, Fratello°(y)))$

4. $\forall x\ \forall y\ ((Female(x) \wedge SpettacoloSportivo(y)) \rightarrow Detesta(x,y))$
5. $\exists y\ \forall x\ ((Female(x) \wedge SpettacoloSportivo(y)) \rightarrow Detesta(x,y))$
6. $\exists y\ SpettacoloSportivo(y) \wedge \forall x\ (Female(x) \rightarrow Detesta(x,y))$

Equivalenza di formule

Queste formule sono equivalenti?
hanno l'identico significato?

1. $\forall x\ \exists y\ ((Female(x) \wedge SpettacoloSportivo(y)) \rightarrow Detesta(x,y))$

2. $\forall x\ \forall y\ ((Female(x) \wedge SpettacoloSportivo(y)) \rightarrow Detesta(x,y))$

3. $\forall y\ (SpettacoloSportivo(y) \rightarrow \forall x\ (Female(x) \rightarrow Detesta(x,y)))$

4. $\exists y\ SpettacoloSportivo(y) \wedge \forall x\ (Female(x) \rightarrow Detesta(x,y))$

Come verifichiamo l'equivalenza di formule?

Applicando regole note di trasformazione
oppure con le tavole di verità

Esempio: quali di queste formule sono equivalenti?

1. $A \wedge B \wedge C$
2. $A \wedge B \rightarrow C$
3. $A \vee B \rightarrow C$
4. $\neg A \vee \neg B \vee C$
5. $\neg (\neg A \vee \neg B \vee \neg C)$

Perché è importante imparare a formalizzare

I sistemi di simulazione di dialoghi e di generazione di messaggi in linguaggio naturale lavorano trasformando in forma simbolica frasi in LN (e viceversa) e ragionando sulla forma simbolica

Formalizziamo conoscenza su domini con un linguaggio del primo ordine

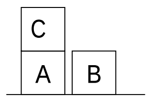
occorre scegliere in modo opportuno
gli elementi del linguaggio
e tradurre correttamente
le frasi in linguaggio naturale
in formule 'valide',
facendo sì che
la loro interpretazione sia 'vera'

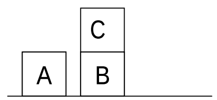
Primo Esempio: Mondo dei blocchi*

On(C,A): C è sopra A
Clear(C): C non ha nessun blocco sopra di lui
Clear(B): B non ha nessun blocco sopra di lui
Table(B): B è sul tavolo
Table(A): A è sul tavolo

Conoscenza *specifica* sullo stato del mondo

Stato attuale del mondo





On(C,B)
Clear(A)

Stato del mondo da raggiungere

* Un dominio semplice, utilizzato spesso per descrivere metodi di IA

Primo Esempio: Mondo dei blocchi

Possiamo rappresentare, con lo stesso linguaggio, le *azioni* che è possibile compiere sui blocchi.

L'operazione di '*Stack di x su y*' consiste nel sovrapporre il blocco x al blocco y; richiede che x sia sulla tavola e che entrambi i blocchi siano clear:

Azione: Stack(x,y)

Precondizioni: Clear(y) $\wedge$ Clear(x) $\wedge$ Table(x)

Effetti: On(x,y) $\wedge \neg$ Clear(y) $\wedge \neg$ Table(x)

L'operazione di '*Unstack di x da y*' è quella inversa:

Azione: Unstack(x,y)

Precondizioni: Clear(x) $\wedge$ On(x,y)

Effetti: $\neg$ On(x,y) $\wedge$ Clear(y) $\wedge$ Table(x)

Conoscenza *generale* sulle azioni
Che si possono effettuare sul mondo

Secondo Esempio: Frasi in linguaggio naturale

Da Es 2.5

Oz: Il mio nome è Valentina

Name(S,Valentina)

Da Es 2.5

Oz: Mangiare a orari fissi ***aiuta ad*** *evitare di saltare i pasti*

$\forall$x (Person(x) $\wedge$ EatAtFixedTime(x)) $\rightarrow$ AvoidJumpMeal(x)*

Da Es 2.3

S: That's United flight 659; it arrives back into San Jose at 6.42

Company(F, United) $\wedge$ Arrives(F, SanJose, 6.42)

Da Es 2.5

Oz: Sono qui per darti dei suggerimenti su come migliorare la tua dieta

Goal(S, Suggest(S, HowToImprove°(diet)))

(nota che questa non è una formula in un linguaggio del prim'ordine! Perché???)

* Vedremo il ruolo dell'incertezza!!

Esercizio

Conoscenza generale:

I cavalli sono, in generale, più veloci dei cani e c'è un levriero che è più veloce di qualsiasi coniglio.

Conoscenza specifica:

Fulmine è un cavallo; Roger è un coniglio.

Domanda:

Fulmine è più veloce di Roger?

Formalizziamo

$\forall$x$\forall$y (Cavallo(x) $\wedge$ Cane(y) $\rightarrow$ PiuVeloce(x,y))

$\exists$y Levriero(y) $\wedge$ $\forall$z (Coniglio(z) $\rightarrow$ PiuVeloce(x,z))

Cavallo(F)

Coniglio(R)

e aggiungiamo la conoscenza generale implicita:

$\forall$x $\forall$y $\forall$z ((PiuVeloce(x,y) $\wedge$ PiuVeloce(y,z)) $\rightarrow$ PiuVeloce(x,z))

$\forall$y (Levriero(y) $\rightarrow$ Cane(y))

Esercizio

Conoscenza generale:

Tutti gli uccelli volano, tranne gli struzzi.

Con l'eccezione delle balene, tutti i mammiferi vivono sulla terra.

Conoscenza specifica:

Pippo è uno struzzo, Bianca è una balena.

Domanda:

Chi dei due vola?

Formalizziamo

$\forall$x ((Bird(x) $\wedge$ $\neg$Ostrich(x)) $\rightarrow$ Flies(x))

$\forall$y ((Mammal(y) $\wedge$ $\neg$Whale(y)) $\rightarrow$ OnGround(y))

Ostrich(P)

Whale(B)

e aggiungiamo la conoscenza generale implicita:

$\forall$x (Ostrich(x) $\rightarrow$ Bird(x))

$\forall$y (Whale(y) $\rightarrow$ Mammal(y))

Ora possiamo porci quesiti del tipo: qual è la sostituzione che rende true $\neg$ Flies(x)?

Riprendiamo a ragionare sull'Es 2.4
L' esempio nel campo della persuasione

- **Task model: le strategie persuasive di Walton**

a. *Appeal to positive consequences*

"Se *ritieni che* compiere una determinata azione comporti conseguenze importanti per te e sei in grado di compierla, dovresti farla".

$\forall x\ \forall a\ \forall g\ \ ((Implies(a,g) \wedge Likes(x,g) \wedge CanDo(x,a)) \rightarrow (ShouldDo(x,a))$

b. *Appeal to negative consequences*

"Se ritieni che compiere una determinata azione comporti conseguenze che desideri evitare e sei in grado di evitarle di compierla, dovresti farlo".

$\forall x\ \forall a\ \forall g\ ((Implies(a,g) \wedge \neg Likes(x,g) \wedge CanAvoid(x,a)) \rightarrow \neg (ShouldDo(x,a))$

- **Domain model**

"Fare sport fa bene alla salute e alla forma fisica."

$\forall s\ Sport(s) \rightarrow Implies(s,GoodHealth)$

$\forall s\ Sport(s) \rightarrow Implies(s,GoodShape)$

"Il running è un particolare tipo di sport". Tutti i giovani senza particolari problemi di salute sono in grado di fare running.

$Sport(R)$

$\forall x\ \forall s\ (Sport(s) \wedge Person(x) \wedge Young(x) \wedge Healthy(x)) \rightarrow CanDo(x,s))$

- **User model**

"Giuseppe è giovane,
non ha particolari problemi di salute
e tiene alla sua forma fisica.
Come molti giovani,
non si preoccupa invece, per ora, della sua salute."

$Young(G)$
$Healthy(G)$
$Likes(G,GoodShape)$
……

Facciamo il lavoro inverso:
Da formule a discorsi

Traduciamo queste formule in buone frasi in linguaggio naturale:

1. $\forall x\ (SiFerma(x) \rightarrow Perduto(x))$
2. $\forall x\ \ (Luccica(x) \rightarrow Oro(x))$
3. $\forall x\ \exists y\ (Padre(y, x) \wedge Litiga(y, Moglie°(y)))$
4. $\forall x\ \exists y\ \exists z\ (Padre(y, x) \wedge Moglie(z, y) \wedge Litiga(y,z))$
5. $\forall x\ \forall y\ (Person(x) \wedge Hungry(x) \wedge Food(y)) \rightarrow ShouldEat(x,y))$

   "Bisognerebbe mangiare solo se si ha fame" (Es 2.5)

6. $\forall x\ HealthyDiet(x) \rightarrow FewFats(x)$

   "Limitare la dose di grassi … rappresenta un elemento fondamentale di una dieta sana" (Es 2.5).

Ragioniamo, ora, sulla conoscenza formalizzata

Come si può fare inferenza?

Da:

$\forall x \forall y\ ((Cavallo(x) \wedge Cane(y)) \rightarrow PiuVeloce(x,y))$

$\exists y\ Levriero(y) \wedge \forall z\ (Coniglio(z) \rightarrow PiuVeloce(x,z))$

Cavallo(F)

Coniglio(R)

$\forall x\ \forall y\ \forall z\ (PiuVeloce(x,y) \wedge PiuVeloce(y,z) \rightarrow PiuVeloce(x,z))$

$\forall y\ (Levriero(y) \rightarrow Cane(y))$

Si può derivare:

PiuVeloce(F,R)? *(domanda di tipo 'yes/no')*

Regole di Inferenza (un esempio!!) (sistema di Genesereth e Nilsson)

MP: modus ponens: $(\varphi \rightarrow \psi) \wedge \varphi \vdash \psi$

MT: modus tollens: $(\varphi \leftarrow \psi) \wedge \neg\psi \vdash \neg\varphi$

AE: and elimination: $(\varphi \wedge \psi) \vdash \varphi\ ,\ \psi$

AI: and introduction: $\varphi\ ,\ \psi \vdash (\varphi \wedge \psi)$

UI: universal instantiation: $\forall x\ \varphi \vdash \varphi$ con x/A

EI: existential instantiation: $\exists x\ \varphi \vdash \varphi$ con x/F°(x_1, x_2,...x_n)

Coniglio(R)

$\exists y\ Levriero(y) \wedge (\forall z\ Coniglio(z) \rightarrow PiuVeloce(x,z))$

EI: $Levriero(L) \wedge (\forall z\ Coniglio(z) \rightarrow PiuVeloce(L,z))$

AE: Levriero(L); $(\forall z\ Coniglio(z) \rightarrow PiuVeloce(L,z))$

UI: $Coniglio(R) \rightarrow PiuVeloce(L,R)$

MP: PiuVeloce(L,R)

$\forall y\ Levriero(y) \rightarrow Cane(y)$

UI: $Levriero(L) \rightarrow Cane(L)$

MP: Cane(L)

Cavallo(F)

$\forall x \forall y\ Cavallo(x) \wedge Cane(y) \rightarrow PiuVeloce(x,y)$

UI: $Cavallo(F) \wedge Cane(L) \rightarrow PiuVeloce(F,L)$

AI: $Cavallo(F) \wedge Cane(L)$

MP: PiuVeloce(F,L)

$\forall x \forall y\ \forall z\ PiuVeloce(x,y) \wedge PiuVeloce(y,z) \rightarrow PiuVeloce(x,z)$

UI: $PiuVeloce(F,L) \wedge PiuVeloce(L,R) \rightarrow PiuVeloce(F,R)$

AI: $PiuVeloce(F,L) \wedge PiuVeloce(L,R)$

MP: PiuVeloce(F,R)

MP: modus ponens
MT: modus tollens
AE: and elimination
AI: and introduction
UI: univ. instantiation
EI: exist. instantiation

1. $\forall x\ Bird(x) \wedge \neg Ostrich(x) \rightarrow Flies(x)$
2. $\forall x\ Bird(x) \wedge Ostrich(x) \rightarrow \neg Flies(x)$
3. $\forall y\ Mammal(y) \wedge \neg Whale(y) \rightarrow OnGround(y)$
4. $\forall y\ Mammal(y) \wedge Whale(y) \rightarrow \neg OnGround(y)$
5. $\forall y\ OnGround(y) \rightarrow \neg Flies(y)$
6. $\forall y\ \neg OnGround(y) \rightarrow Flies(y)$
7. $\forall x\ Ostrich(x) \rightarrow Bird(x);$
8. $\forall y\ Whale(y) \rightarrow Mammal(y)$
9. $Ostrich(P)$; 10. $Whale(B)$

MP: modus ponens
MT: modus tollens
AE: and elimination
AI: and introduction
UI: univ. instantiation
EI: exist. instantiation

====================

$\forall x\ Ostrich(x) \rightarrow Bird(x);$ UI: Ostrich(P) $\rightarrow$ Bird(P);
$Ostrich(P)$ MP: Bird(P)
$\forall x\ Bird(x) \wedge Ostrich(x) \rightarrow \neg Flies(x)$
UI: Bird(P) $\wedge$ Ostrich(P) $\rightarrow \neg$ Flies(P)
AI: Bird(P) $\wedge$ Ostrich(P); MP: $\neg$ Flies(P)
$\forall y\ Whale(y) \rightarrow Mammal(y)$ UI: Whale(B) $\rightarrow$ Mammal(B)
MP: Mammal(B)
$\forall y\ Mammal(y) \wedge Whale(y) \rightarrow \neg OnGround(y)$
UI: Mammal(B) $\wedge$ Whale(B) $\rightarrow \neg$ OnGround(B)
AI: Mammal(B) $\wedge$ Whale(B) ; MP: $\neg$ OnGround(B)
$\forall y \neg OnGround(y) \rightarrow \neg Flies(y)$
UI: $\neg$OnGround(B) $\rightarrow \neg$Flies(B); MP: $\neg$Flies(P)

Ci sono molti altri modi di fare inferenza

Ne studieremo, in particolare, uno:

Quello basato sul

Principio di Risoluzione.

Ricerca delle soluzioni nello spazio degli stati

Due direzioni possibili:

Si parte dall'obiettivo che s'intende raggiungere ('stato goal') e si applicano gli inversi degli 'operatori' di trasformazione disponibili, fino ad arrivare ad uno stato che corrisponde allo 'stato iniziale (*backward*).

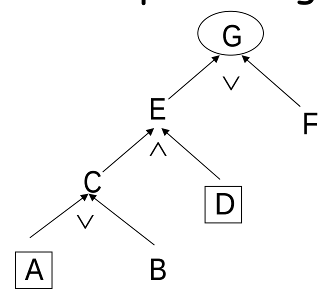
Si parte dallo stato iniziale, e si applicano gli operatori fino a raggiungere lo stato 'goal' (*forward*).

Ricerca delle soluzioni nello spazio degli stati

$A \vee B \rightarrow C$

$C \wedge D \rightarrow E$

$E \vee F \rightarrow G$



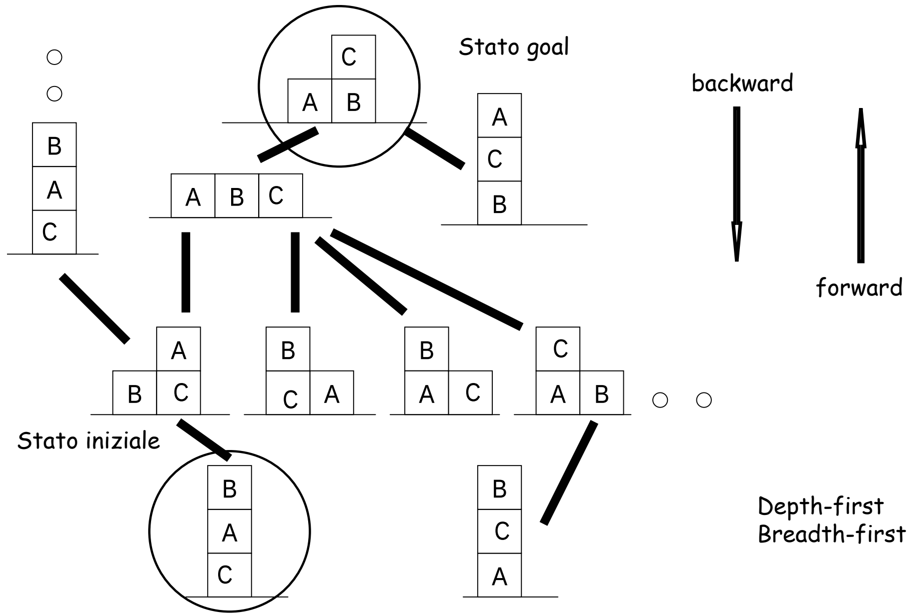
Stato iniziale: A, D

Goal: G

Ricerca backward: G?; E? or F?; E?: C? and D?; D; C?: A? or B?; A
G = T

Ricerca forward: A; C; D; E; G
G = T

Ricerca delle soluzioni: un altro esempio



Ricerca delle soluzioni nello spazio degli stati

Vedremo come il Principio di Risoluzione permetta di simulare i due metodi di ricerca

e come la convenienza del metodo dipenda dalla struttura del problema.

Esempio: