

**Come ragionare sugli stati mentali?
una estensione del Principio di Risoluzione:
la 'Regola di Attachment'**

Bel A $\varphi_1 \vee \psi_1$

Bel A $\varphi_2 \vee \psi_2$

.....

Bel A $\varphi_n \vee \psi_n$

\neg **Bel A** $\varphi_{n+1} \vee \psi_{n+1}$

$\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \varphi_{n+1}$

$\psi_1 \vee \psi_2 \vee \dots \vee \psi_{n+1}$

Che ipotesi?

L'ipotesi alla base della Regola di Attachment è che
l'agente rappresentato ragioni in modo 'coerente'.

Cioè che, se $\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \varphi_{n+1}$
se l'agente crede nei primi n φ_1 ,
non può non credere anche
nella loro conseguenza φ_{n+1}

Aggiungendo φ_{n+1}
all'insieme $\psi_1, \psi_2, \dots, \psi_{n+1}$
si introduce una contraddizione
e si può, quindi, derivare la clausola
vuota.

Come applico la regola di attachment?

Esamino il set di clausole e verifico se esiste almeno una coppia
che contiene letterali di tipo Bel;

Se così non è, la regola non è applicabile

Altrimenti, verifico se almeno uno dei letterali è negativo.

Se così non è, la regola non è applicabile

Altrimenti: isolo dalle clausole i letterali di tipo Bel

Verifico (con il principio di risoluzione) se è applicabile, ad un loro
sottoinsieme, l'inferenza:

$\varphi_1 \wedge \varphi_2 \wedge \dots \wedge \varphi_n \rightarrow \varphi_{n+1}$

con φ_{n+1} argomento del letterale negativo e gli altri, argomenti del
letterale positivo.

Se così non è la regola non è applicabile

Altrimenti derivo, dall'insieme di clausole isolate, la conclusione

$\psi_1, \psi_2, \dots, \psi_{n+1}$

Ripeto per tutti gli insiemi isolabili, e infine formo la congiunzione
delle conclusioni derivate per ogni sottoinsieme.

ψ_1, ψ_2, ψ_3 **Formalizziamo e ragioniamo**

1. **Corvo(Sk)**, **(Bel J $(\forall x \text{Corvo}(x) \rightarrow \text{Black}(x))$)** φ_1
2. **$\neg \text{Black}(Sk)$** , **(Bel J $(\forall x \text{Corvo}(x) \rightarrow \text{Black}(x))$)** φ_2
3. **$\neg \text{Corvo}(A)$** , **(Bel J $\text{Corvo}(A)$)** φ_3
4. **$\neg \text{Bel J Black}(A)$** φ_4

Possiamo applicare la regola di attachment alle clausole 2,3,4,
ottenendo:

$\neg \text{Black}(Sk) \vee \neg \text{Corvo}(A)$,

Oppure alle 1,3,4, ottenendo:

$\text{Corvo}(Sk) \vee \neg \text{Corvo}(A)$

Poiché entrambe le conclusioni sono valide, abbiamo:

$(\neg \text{Black}(Sk) \vee \neg \text{Corvo}(A)) \wedge (\text{Corvo}(Sk) \vee \neg \text{Corvo}(A))$ che
equivale a:

$\neg \text{Corvo}(A) \vee (\text{Corvo}(Sk) \wedge \neg \text{Black}(Sk))$

"A non è un corvo, oppure non tutti i corvi sono neri"

Nell'esempio precedente, potevo isolare due sottoinsiemi di clausole ai quali è applicabile la regola di attachment:

L'insieme 1, 3 e 4 e

l'insieme 2, 3 e 4

In entrambi i casi, la clausola 4 conteneva un letterale Bel negativo e le altre un letterale Bel positivo.

In entrambi i casi, le clausole contenevano anche letterali di tipo ψ_1

Dal primo sottoinsieme ho derivato una conclusione:

$\neg \text{Black}(\text{Sk}) \vee \neg \text{Corvo}(\text{A}),$

Dal secondo, un'altra:

$\text{Corvo}(\text{Sk}) \vee \neg \text{Corvo}(\text{A})$

E le ho poste in 'and' fra loro (perché entrambe sono derivabili dall'insieme di clausole iniziale, con diverse applicazioni della regola!!)

Quesito 'intrigante' sulle Relazioni Retoriche

T1: Saresti dovuto venire al party per la laurea di Francesco.

T2: Avresti incontrato tutti i tuoi amici di liceo.

Con che struttura retorica si descrive? Non può essere una Motivation vista la presenza dell'azione al passato?

No, e non se che RR possa essere.

Una Background? Una CauseEffect????

Quesiti 'implacabili' sui Metodi di NLG

Quesito:

Non mi è chiaro come si possa risolvere l'esercizio dei blocchi rossi e verdi a partire dallo stato iniziale in cui ci sono 2 blocchi rossi A e B uno sopra all'altro e un blocco verde C per terra e avendo a disposizione solo 4 operatori che permettono di fare:

unstack di un blocco rosso da uno verde,

unstack di un blocco verde da uno rosso,

stack di un blocco rosso su uno verde

stack di un blocco verde su uno rosso.

Secondo me c'è bisogno per forza di un operatore che permetta di fare unstack di un blocco rosso da un blocco rosso, senza di esso non sono riuscita a risolvere l'esercizio.

Come si può risolvere altrimenti?

... È giusto!

Quesito:

Nella slide 17 'Direzione della ricerca della soluzione' non mi è chiaro il passaggio di unificazione delle righe 4 e 5.

In altre parole, secondo me dovrebbe essere:

not(T(On(A,y)S1)), not(T(Clear(A,y)S1)),
Do(Unstack(A,y)S1),

Ans(Unstack(A,y)).

Quindi manca l'unificazione di s\S1 oltre a quella di x\A perchè poi comunque S1 viene sostituito al posto di s nelle clausole, il termine
Do(Unstack(A,y),S1),

E poi non capisco perché lei ha scritto

(Ans(Unstack(A,y),S1): da dove viene questo S1 in Ans?

Direzione della ricerca della soluzione (lucido rivisto)

Ricerca della soluzione backward,
a partire dalla descrizione del goal:

- 1 {T(On(A,B), S₁)}
- 2 {T(Clear(A), S₁)}
- 3 {T(On(x,y),Do(Stack(x,y),s)),¬T(Table(x),s),¬T(Clear(x),s),
¬T(Clear(y), s))}
- 4 {T(Table(x), Do(Unstack(x,y),s)), ¬T(On(x,y),s), ¬T(Clear(x), s))}
- 5 {¬ T(Table(A), Do(a, S₁)), Ans(a)}

In questo caso:

(4,5) con x/A, s/S1 e a/Unstack(A,y),S1 produce:

{¬T(On(A,y),S₁),¬T(Clear(A),S₁),Ans(Unstack(A,y),S₁), 6

(1,6) con y/B produce:

{¬T(Clear(A),S₁),Ans(Unstack(A,B),S₁), 7

(2,7) produce {Ans(Unstack(A,B))}

Quando si parla, nella slide 40, degli 'Operatori per la pianificazione del discorso' si dice che

Header corrisponde alla descrizione dell'obiettivo dell'operatore,

Effects sono i suoi effetti sulla mente di H e

Constraints sono vincoli su S e H che devono essere soddisfatti per poter applicare l'operatore.

Nella slide successiva poi si parla degli operatori di Moore e Paris e di dice che l'effetto è l'obiettivo dell'azione descritta dall'operatore. Ma allora questi 2 tipi di operatori sono diversi?

In quale di queste due classi rientrano quindi gli Operatori Multimodali di Maybury (slide 44)?

Nell'esempio fatto sull'operatore Identify-location-linguistically(slide 44), nelle Constraints c'è come vincolo Entity(e). Ma non avevamo detto che le Constraints erano VINCOLI SU S e H che devono essere soddisfatti per poter applicare l'operatore?

E ancora, si parla di Constraints e Precondition, ma non erano la stessa cosa (almeno così sembra nella slide 41)? Se così non fosse qual'è la differenza tra Constraints e Precondition?

Operatori di Moore e Paris

Esempio: operatore di piano per la MOTIVATION

- EFFECT: (MOTIVATION ?act ?goal)
- CONSTRAINTS: (AND (STEP ?act ?goal)
(GOAL ?hearer ?goal))
- NUCLEUS: (BEL ?hearer (STEP ?act ?goal))
- SATELLITES: NIL

Operatori Multimodali di Maybury (1)

=====

HEADER: Identify(S,H, e)
CONSTRAINTS: Entity(e)
PRECONDITIONS: Goal S, KnowAbout(H, Location°(e))
EFFECTS: KnowAbout(H, Location°(e))
DECOMPOSITION: Inform(S,H, Location°(e))

=====

NAME: Identify-location-linguistically&visually
HEADER: Identify(S,H, e)
CONSTRAINTS: CartographicEntity(e)
PRECONDITIONS: Visible(e) ∧ Goal S, KnowAbout(H, Location°(e))
EFFECTS: KnowAbout(H, Location°(e))
DECOMPOSITION: IndicateDeictically(S,H, Location°(e)) ∧ Inform(S,H, Location°(e))

=====

Operatori Multimodali di Maybury (3)

NAME: extended-description
HEADER: Describe (S,H,e)
CONSTRAINTS: Entity (e)
PRECONDITIONS:
ESSENTIAL: KnowAbout(S,e) ∧ Want(S, KnowAbout(H,e))
DESIRABLE: ¬KnowAbout(H,e)
EFFECTS: KnowAbout(H,e)
DECOMPOSITION: Define(S,H,e)
optional: Detail(S,H,e) ∧ Divide(S,H,e) ∧ Illustrate(S,H,e) ∧ Give-Analogy(S,H,e)

Il goal Describe(S,H,e) si decompone nei sub-goal Define(S,H,e) e (opzionalmente) Detail(S,H,e), Illustrate(S,H,e) e Give-Analogy/S,H,e).
 Ha, come precondizioni, che S conosca l'entità da descrivere e che voglia che anche H la conosca. Quest'ultimo è l'effetto prodotto dall'operatore.

Gli operatori non hanno tutti la stessa struttura, e i pianificatori non hanno tutti la stessa logica di funzionamento

Maybury distingue fra:

Constraints: se non sono vere, l'operatore non è applicabile e

Preconditions: se non sono vere, si cerca un altro operatore che le rende vere (il cui slot 'Effect' è unificabile con lo slot 'Precondition' o una sua parte).

Inoltre, pianifica unificando Header con Decomposition.

Applica, quindi, un doppio processo di unificazione: Constraint-Preconditions e Header-Decomposition

