

Pianificazione del Discorso

Riassumendo:

1. **Discorso:** Un discorso (testo) è un set strutturato di clausole. A causa delle relazioni presenti fra queste clausole, esse possono essere raggruppate in segmenti. La struttura del discorso è espressa dall'innestamento di segmenti in base alle relazioni specifiche che li governano. Un discorso può quindi essere rappresentato come un albero in cui ogni nodo è radice di un sottoalbero. A livello più alto è presente un solo nodo detto radice. A livello delle foglie ci sono segmenti primitivi (non più scomponibili) che corrispondono a singole clausole grammaticali o a atti comunicativi primitivi.
2. **Purpose:** Ogni segmento del discorso ha uno scopo ed è rappresentato in ogni nodo dell'albero. Quindi, ogni segmento può essere visto come un passo del piano che consente di raggiungere il goal comunicativo globale del discorso.
3. **Coherence:** organizzazione del testo che viene rafforzata dall'uso dei vincoli imposti dalle relazioni fra segmenti (i.e. RST).

Pianificazione del Discorso

Riassumendo:

4. **Discourse segment:** un segmento del discorso S è rappresentato dalla tupla: *nome scopo contenuto*
dove:
 1. il **nome** è un identificatore univoco per S
 2. lo **scopo** corrisponde ad uno o più goal comunicativi dello Speaker nei confronti dello stato mentale di chi ascolta;
 3. Il **contenuto** è
 - Una lista ordinata di segmenti del discorso Si uniti da relazioni (RR)
 - Un singolo segmento
 - Contenuto semantico da comunicare, in genere, in una sola frase.
5. **Discourse structure:** la struttura del discorso D è un segmento del discorso che non è contenuto in nessun altro segmento e tutte le sue foglie contengono materiale semantico da comunicare.

RST e Discourse Planning

Vari approcci all'operazionalizzazione della RST per generare dinamicamente un discorso.

In Hovy, un operatore per pianificare un discorso è definito come una tupla:

nome effetti vincoli precondizioni scomposizione

Dove:

il nome è l'identificatore univoco del segmento

gli effetti corrispondono ad uno piu' goal comunicativi che il piano raggiunge se eseguito in maniera appropriata.

i vincoli sono fatti nella base di conoscenza o nello user model che devono essere veri prima di poter applicare l'operatore

Le precondizioni sono fatti nella base di conoscenza o nello user model che dovrebbero essere verificati per ottenere una comunicazione "efficace"

La scomposizione è una lista ordinata di sottogoal da soddisfare. I sottogoal sono, in genere di due tipi: intenzionali o linguistici.

Operatori di Moore e Paris

Name: persuadebymotivation

Effect: (PERSUADED H (GOAL H (do H ?act)))

Constraints: AND((GOAL S ?g)

(GOAL H ?g)

(STEP ?act ?g))

Nucleus: (FORALL ?g (MOTIVATION ?act ?g))

Satellites:

Per raggiungere uno stato

in cui H è persuaso a

compiere l'azione ?act.

SE ?act è un passo per

raggiungere un goal g di H

e g e' anche un goal di S

ALLORA motiva ?act in

termini di questi goal(s).

Name: motivateactbymeans

Effect: (MOTIVATION ?act ?goal)

Constraints: AND((GOAL S ?goal)

(GOAL H ?goal)

(STEP ?act ?goal))

Nucleus: (Inform S H (GOAL G goal))

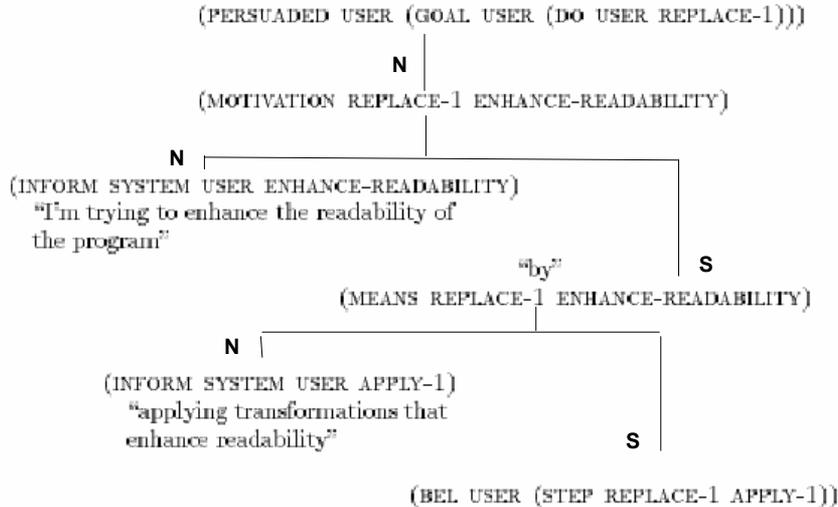
Satellites: (means ?goal ?act)

Per raggiungere uno stato

in cui H è motivato a

compiere ?act.

Operatori di Moore e Paris



Operatori di Maybury (90')

NAME:	Extended-description
HEADER:	Describe(S, H, entity)
CONSTRAINTS:	Entity?(entity)
PRECONDITIONS	
ESSENTIAL:	KNOW-ABOUT(S, entity) \wedge WANT(S, KNOW-ABOUT(H, entity))
DESIRABLE:	\neg KNOW-ABOUT(H, entity)
EFFECTS:	KNOW-ABOUT(H, entity)
DECOMPOSITION:	Define(S, H, entity) <i>optional</i> (Detail(S, H, entity)) <i>optional</i> (Divide(S, H, entity)) <i>optional</i> ((Illustrate(S, H, entity)) \vee Give-Analogy(S, H, entity))

(S and H stand for Speaker and Hearer respectively. Describe, Define, Detail, Divide, Illustrate, and Give-analogy are communicative intentions.)

Esempio

Vogliamo riprodurre il brano del manuale di *vi*:

vi is a display oriented text editor based on ex. ex and vi are, in fact, the same text editor; it is possible to get to the command mode of ex from within vi and vice-versa.

Supponiamo che la base di conoscenza includa i fatti:

- `is-a (vi, text editor)`
- `is-a (ex, text editor)`
- `attribute (vi, display oriented)`
- `based-on (vi, ex)`
- `same (vi, ex)`
- `get-from (vi, ex)`
- `get-from (ex, vi)`

Azione basica: supponiamo che le azioni di base del nostro generatore siano le *Inform*: ogni volta che una *Inform* è applicata ad un argomento *X* una frase viene

Esempio

Azione basica: supponiamo che le azioni di base del nostro generatore siano le *Inform*: ogni volta che una *Inform* è applicata ad un argomento *X* una frase viene generata (ignoriamo per il momento come).

Per esempio:

- `Inform (is-a (vi, text-editor))` produce:
“vi is a text editor”
- `Inform (get-from (vi, ex))` produce:
“it is possible to get to the command mode of ex from within vi”.

Operatori:

- GOAL: Explain ?x
- CONSTRAINTS: -
- SUBGOALS: Describe ?x
- DescribeAnalogies ?x
- RR: Elaboration

- GOAL: Describe ?x
- CONSTRAINTS: Exists ?y based-on(?x,?y)
- Know(Hearer,?y)
- SUBGOALS: Define ?x
- Inform based-on(?x,?y)
- RR: Background

- GOAL: Define ?x
- CONSTRAINTS: Exists ?y is-a(?x,?y)
- SUBGOALS: Inform is-a(?x,?y)
- Forall ?a attribute(?x,?a) Inform attribute(?x,?a)
- RR: Elab Object-Attribute

- GOAL: DescribeAnalogies ?x
- CONSTRAINTS: Exists ?y same(?x,?y)
- SUBGOALS: Inform same(?x,?y)
(Forall ?rel ?rel(?x,?y)
Inform ?rel(?x,?attr)
Forall ?rel ?rel(?y,?x)
Inform ?rel(?y,?x))
- RR: Evidence

Input:(Explain vi)

Testo prodotto:

- *vi is a text editor. vi is display oriented. vi is based on ex. ex and vi are the same. In fact, it is possible to get to the command mode of ex from within vi and it is possible to get to the command mode of vi from within ex.*

Pro e Contro dell'approccio a planning

- **Vantaggi:**

1. I pianificatori sono più flessibili: possono essere modificati facilmente se deve essere aggiunta nuova informazione.
2. Ad ogni passo di decomposizione, il pianificatore può tener traccia del motivo per cui una decisione viene presa: può quindi effettuare backtracking, o giustificare il suo operato.
3. Gli operatori di planning possono essere meglio innestati per dire la stessa cosa in molti modi diversi.

- **Svantaggi:**

1. E' necessaria una rappresentazione più complessa, sia del dominio sia della conoscenza linguistica.
2. Computazionalmente più complesso.
3. Il testo prodotto può essere "peggiore" di quello ottenuto dagli schemi, in termini di naturalezza: spesso un nuovo passo di rifinitura è necessario.