



Analisi Semantica nel Trattamento Automatico del Linguaggio Naturale

Corso di Intelligenza Artificiale: Trattamento Automatico del Linguaggio Naturale
Professore: Amedeo Cappelli
Presentazione: Luisa Colucci

1

Analisi semantica

- Cercheremo di capire come viene realizzata l'analisi semantica all'interno del processo di trattamento automatico del linguaggio
- A tal proposito presenteremo due diverse tecniche per affrontare il processo di analisi semantica:

- 1) LUNAR SYSTEM
- 2) UNIFICATION GRAMMAR

2

Lunar System

Il Lunar System fu sviluppato come supporto per il centro spaziale della NASA (1971).

Il suo compito: permettere ai geologi di fare facilmente richieste sulla composizione chimica delle rocce lunari (risultato della missione Apollo 10).

Idea: permettere ai geologi di accedere con richieste in linguaggio naturale (inglese) al data base della NASA senza dover imparare alcun linguaggio di programmazione e senza essere a conoscenza delle convenzioni usate per la realizzazione del data base.

Tipi di domande:

"What is the average concentration of aluminum in high-alkali rocks?"

"Give me all lunar samples with magnetite"

3

Lunar System: Architettura

Il Lunar System consiste di tre componenti principali:

- **Una grammatica 'general purpose' e un parser** per un largo sottoinsieme dell'inglese;
- **Un semantic-framework** che deve essere in grado, data la rappresentazione sintattica di una frase, di restituire il corrispondente significato;
- **Una componente per memorizzare, manipolare e recuperare i dati** presenti nel data base.

4

Semantica in Lunar

Il **SEMANTIC-FRAMEWORK** di un linguaggio naturale consta, essenzialmente, di tre parti:

- **Un meaning representation language (MRL):** un formalismo per la rappresentazione semantica del significato della frase;
- **Una specifica della semantica della notazione** del MRL (specifica di ciò che le sue espressioni significano);
- **Una procedura di interpretazione semantica:** una procedura per la rappresentazione semantica appropriata di una data frase del linguaggio naturale.

5

Meaning Representation Language

Caratteristiche:

- deve essere capace di rappresentare con precisione, formalmente e senza ambiguità il significato che si può dare ad una frase;
- dovrebbe facilitare la traduzione algoritmica della frase in linguaggio naturale nella corrispondente rappresentazione semantica;

6

Lunar MRL

Si basa, essenzialmente, su tre tipi di costruttori:

DESIGNATORS: denotano oggetti o classi di oggetti del data base.

PROPOSITIONS: corrispondono ad asserzioni che possono vere o false sul data base e sono formati da predicati che hanno designator come argomenti.

COMMANDS: veri e propri comandi utilizzati per svolgere azioni

7

Designator

- **Specificatori individuali:** nomi propri e variabili

Esempi: **S10046** è il designatore per un particolare campione

OLIV è il designatore per un particolare minerale (Olivine)

X_i è una variabile che può denotare un qualsiasi oggetto del DB

- **Specificatori di classi:** denotano nomi di classi di elementi su cui opera la quantificazione. Essi sono formati da una funzione di enumerazione per classi più possibili argomenti

Esempi: **(SEQ TYPECS)** specificazione della classe delle rocce di tipo C

(DATA LINE S10046 OVERALL OLIV) specificazione dell'insieme delle linee della tavola chimica del campione S10046 che corrispondono alla tripla sample/phase/constituent specificata

8

Proposition

Predicati che hanno designator come argomenti.

(CONTAIN S10046 OLIV)

All'interno delle proposition é possibile utilizzare anche connettivi logici.

(AND (CONTAIN X₃ OLIV) (NOT (CONTAIN X₃ PLAG)))

Command

Operatori con argomenti.

(TEST (CONTAIN S10046 OLIV)

(PRINTOUT (...))

9

Quantification

La funzione di quantificazione FOR rappresenta uno dei maggiori poteri espressivi del MRL.

(FOR <quant> X / <class> : (pX) ; (qX))

<quant> specifico quantificatore es. EVERY,SOME.....

X variabile della quantificazione

<class> insieme sul quale la quantificazione opera (range di appl.)

(pX) proposizione che restringe il range

(qX) espressione che si sta quantificando (comando o propos.)

(FOR EVERY X₁ /(SEQ SAMPLES) :

(CONTAIN X₁ SILICON) ; (PRINTOUT X₁))

10

Specifica della semantica della notazione

•Come già detto, la componente di interpretazione semantica del linguaggio naturale non necessita solo di una **NOTAZIONE** per la rappresentazione del significato di una frase, necessita anche di una **SPECIFICA SEMANTICA DELLA NOTAZIONE**.

•Questa semantica della notazione, in Lunar é data:

**associando alla notazione delle procedure
che possono essere eseguite**

11

•Così facendo, il nostro **Meaning Representation Language** é utilizzato:

- rappresentazione "dichiarativa"
- specifica di procedure che possono essere direttamente eseguite.

CONTAIN (S10046 OLIV) 

•Ad ogni **PREDICATO** é associata una procedura che determina il valore di verità del predicato applicato ai suoi argomenti.

• Ad ogni **FUNZIONE** é associata una procedura che calcola il valore assunto dalla funzione per i dati valori dei suoi argomenti.

• Ad ogni **SPECIFICATORE INDIVIDUALE** é associata una procedura che restituisce un puntatore ad un appropriato oggetto appartenente al DB.

.....

12

Interpretazione Semantica

- Siamo arrivati al punto finale: **trovare un modo per assegnare il significato ad una frase.**
- Questa fase si basa su un insieme di regole di interpretazione semantica del tipo *pattern* → *action*.
- Tali regole permettono di passare dalla struttura sintattica restituita dal parser ad una struttura contenente il significato della frase.
- Data la seguente frase

“S10046 contains silicon”

Il parser (di tipo ATN) restituisce come risultato la seguente struttura sintattica:

```

S      DCL
  NP   NPR S10046
  AUX  TNS PRESENT
  VP   V   CONTAIN
        NP NPR SILICON
    
```

13

La regola *pattern* → *action* utilizzata per l'interpretazione semantica della precedente frase é la seguente:

[S : CONTAIN

(S.NP (MEM 1 SAMPLE))

(S.V (OR (EQU 1 HAVE)

(EQU 1 CONTAIN))

(S.OBJ (MEM 1 (ELEMENT OXIDE ISOTOPE)))

→ (QUOTE (CONTAIN (# 1 1) (# 3 1)))]

La **PARTE SINISTRA** della regola é formata da tre “templates”. Fra i templates e la struttura sintattica si effettua un'operazione di matching. In particolare:

- **IL PRIMO TEMPLATE** richiede che la frase che stiamo interpretando abbia come soggetto della Noun Phrase un elemento della classe semantica SAMPLE.
- **IL SECONDO TEMPLATE** richiede che il verbo sia “have” oppure “contain”
- **IL TERZO TEMPLATE** richiede un complemento oggetto che appartenga alla classe semantica element o oxide o isotope.

14

[S : CONTAIN

(S. NP (MEM 1 SAMPLE))

(S.V (OR (EQU 1 HAVE)

(EQU 1 CONTAIN))

(S.OBJ (MEM 1 (ELEMENT OXIDE ISOTOPE)))

→ (QUOTE (CONTAIN (# 1 1) (# 3 1)))]

La **PARTE DESTRA** della regola ci dice che una struttura sintattica che ha i requisiti specificati nella parte sinistra può essere interpretata come un'istanza dello schema **CONTAIN (X, Y)**.

In particolare **(CONTAIN (# 1 1) (# 3 1))**

Rappresenta sia il nome del predicato contain del MRL che la procedura che determina il valore di verità.

Interpretazione del complemento oggetto presa nel nodo 1 del template 3.

Interpretazione del soggetto presa nel nodo 1 del template 1.

15

L'unificazione nelle grammatiche

- Negli ultimi anni si é avuto un grandissimo sviluppo di nuovi formalismi grammaticali basati sul concetto di **UNIFICAZIONE**: Functional Unification Grammar, Lexical Functional Grammar.
- **Caratteristiche comuni**:
 1. **phrase-structure grammars**: assegnano ad ogni frase un albero strutturato delle sue componenti;
 2. **monotonia**: il processo di interpretazione può solamente aggiungere informazione alle strutture già presenti e non può assolutamente modificarle;
 3. **unificazione**: operazione che confronta coppie di espressioni e determina se queste possono essere la descrizione dello stesso oggetto

caso positivo → costruisce una nuova espressione più specifica di quella precedente con i dettagli delle componenti

16

Simple Unification Grammar

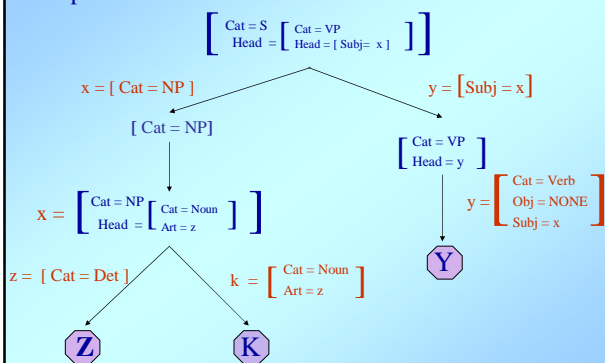
- Formalismo basato sull'utilizzo di *functional descriptor* (FD).
 - Un *FD* é un oggetto formato da una lista di coppie **<attributo-valore>**
 - attributo: caratteristiche dell'oggetto in esame
 - valore: valore associato all'attributo (atoms-FD)
 - La grammatica é formata da un insieme di regole separate da una freccia:
 - sinistra: functional descriptor
 - destra: 1. *FD*
 - 2. **Constituent**: sequenze di paths
 - 3. **Order**: ordine di lettura delle sequenze di paths
- path: · elenco del nome di attributi il cui valore deve essere unificato
 · < a b c > avrà il valore β se [a = [b = [c = β]]

17

1. $S \rightarrow NP VP$	1. $\left[\begin{array}{l} \text{Cat} = S \\ \text{Head} = \left[\begin{array}{l} \text{Cat} = VP \\ \text{Head} = [\text{Subj} = [\text{Cat} = NP]] \end{array} \right] \end{array} \right]$ → < Head Head Subj > < Head >
2. $VP \rightarrow V NP$	2. $\left[\begin{array}{l} \text{Cat} = VP \\ \text{Head} = \left[\begin{array}{l} \text{Cat} = \text{Verb} \\ \text{Obj} = [\text{Cat} = NP] \end{array} \right] \end{array} \right]$ → < Head > < Head Obj >
3. $VP \rightarrow V$	3. $\left[\begin{array}{l} \text{Cat} = VP \\ \text{Head} = \left[\begin{array}{l} \text{Cat} = \text{Verb} \\ \text{Obj} = \text{NONE} \end{array} \right] \end{array} \right]$ → < Head >
4. $NP \rightarrow \text{Det } N$	4. $\left[\begin{array}{l} \text{Cat} = NP \\ \text{Head} = \left[\begin{array}{l} \text{Cat} = \text{Noun} \\ \text{Art} = [\text{Cat} = \text{Det}] \end{array} \right] \end{array} \right]$ → < Head Art > < Head >
5. $NP \rightarrow N$	5. $\left[\begin{array}{l} \text{Cat} = NP \\ \text{Head} = \left[\begin{array}{l} \text{Cat} = \text{Noun} \\ \text{Art} = \text{NONE} \end{array} \right] \end{array} \right]$ → < Head >

CONTEX-FREE G. TRADUZIONE NEL NUOVO FORMALISMO

Utilizzeremo la precedente grammatica per generare una semplice frase:



19

Ora vediamo possibili *entry* per il nostro esempio:

Z = [Cat = Det]

$\left[\begin{array}{l} \text{Cat} = \text{Det} \\ \text{Word} = \text{the} \end{array} \right]$ $\left[\begin{array}{l} \text{Cat} = \text{Det} \\ \text{Word} = \text{this} \\ \text{Num} = \text{sing} \end{array} \right]$ $\left[\begin{array}{l} \text{Cat} = \text{Det} \\ \text{Word} = \text{these} \\ \text{Num} = \text{plur} \end{array} \right]$

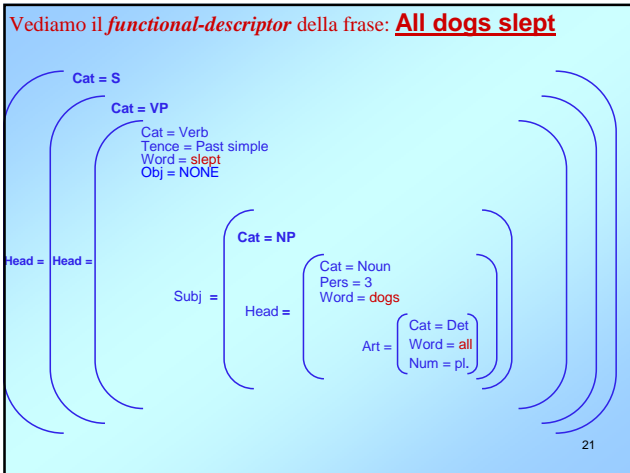
K = [Cat = Noun]

$\left[\begin{array}{l} \text{Cat} = \text{Noun} \\ \text{Word} = \text{dog} \\ \text{Art} = [\text{Num} = \text{sing}] \end{array} \right]$ $\left[\begin{array}{l} \text{Cat} = \text{Noun} \\ \text{Word} = \text{dogs} \\ \text{Art} = [\text{Num} = \text{plur}] \end{array} \right]$

Y = [Cat = Verb]

$\left[\begin{array}{l} \text{Cat} = \text{Verb} \\ \text{Word} = \text{slept} \\ \text{Obj} = \text{NONE} \end{array} \right]$ $\left[\begin{array}{l} \text{Cat} = \text{Verb} \\ \text{Word} = \text{devoirs} \\ \text{**Obj} = [\text{Cat} = NP] \\ \text{Subj} = [\text{Num} = \text{sing}] \end{array} \right]$

20



- ### Vantaggi delle Unification Grammars
- Maggiore potere espressivo rispetto alle contex-free grammars (è possibile scrivere una grammatica per generare il linguaggio $a^n b^n c^n$);
 - L'utilizzo dei descrittori funzionali gioca un ruolo cruciale perché:
 1. ci permette, attraverso l'uso dei suoi attributi, di memorizzare la struttura di trasformazione grammaticale
 2. ci permette, attraverso l'introduzione di nuovi attributi, di aggiungere nuove componenti di descrizione al nostro oggetto
- 22

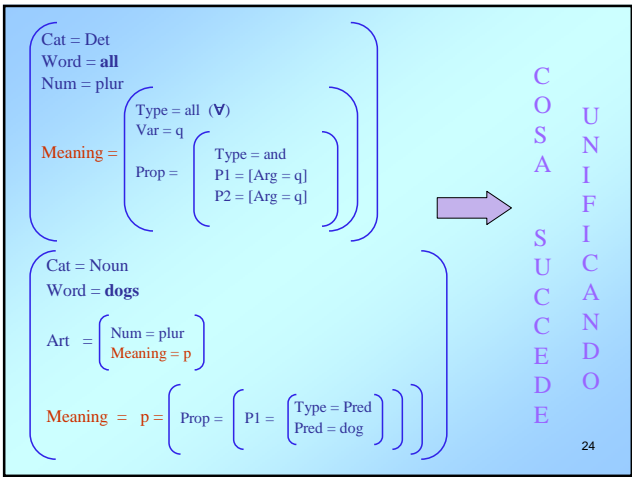
Analisi semantica

Sfruttando la possibilità di aggiungere informazione nei *functional descriptor* riusciamo ad avere un formalismo che ci permette di trattare contemporaneamente analisi sintattica e analisi semantica.

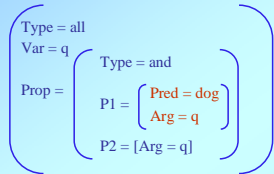
Aggiungiamo un nuovo attributo ai nostri descrittori funzionali dal nome **MEANING**.

Vediamo come cambiano i valori di ingresso per "all" e "dogs".

23



Il risultato dell'unificazione relativo all'attributo meaning é il seguente:



Possiamo vederlo come la descrizione della seguente espressione logica:

$$\forall q. \text{Dog}(q) \wedge \text{P2}(q)$$



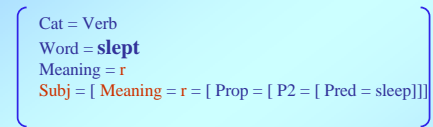
Il predicato P2 deve essere ancora specificato.

QUANDO VERRA' SPECIFICATO?

25

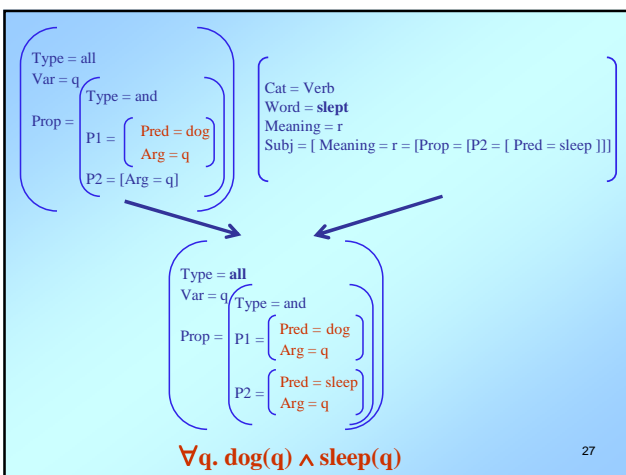
Il predicato P2 sarà specificato quando il gruppo nominale (NP) "all-dogs" diventerà il soggetto (Subj) del verbo "slept".

La struttura di entrata per il verbo "slept" sarà la seguente:



Nella struttura vediamo subito che il *meaning* del verbo (r) deve essere unificato con il *meaning* del soggetto.

26



27

Bibliografia

- W. A. Woods, *Progress in natural language understanding – An application to lunar geology*.
- W. A. Woods, *Semantic and quantification in natural language question answering*, BBN Report 3687, Cambridge, Mass., 1977.
- Amedeo Cappelli, *Appunti sul Lunar System*.
- Martin Kay, *Computational Linguistic = Generalized Unification + Applied Graph Theory*, California.
- Martin Kay, *Unification in Grammar*, Stanford University 1985.

28