

Grammatica, Albero di Derivazione e Forma Logica per Frasi in Italiano

Definizione del Problema

Dato un insieme di frasi semplici in Italiano, come:

- *Il gatto mangia il topo.*
- *La ragazza legge un libro.*

l'obiettivo è:

1. Definire una grammatica che accetti tali frasi.
2. Costruire un sistema di riscrittura in DCG (*Definite Clause Grammar*) per produrre l'albero di derivazione in forma parentetica.
3. Implementare un sistema di riscrittura che determini la forma logica basata su un lessico di predicati verbali e forme nominali semplici.

—

1. Grammatica che Accetta le Frasi

La grammatica è rappresentata in forma BNF come segue:

$S \rightarrow NP VP$ (Una frase è composta da un sintagma nominale e uno verbale)
 $NP \rightarrow Det N$ (Un sintagma nominale è un determinante seguito da un nome)
 $VP \rightarrow V NP$ (Un sintagma verbale è un verbo seguito da un sintagma nominale)
 $Det \rightarrow "il" \mid "la" \mid "un" \mid "una"$
 $N \rightarrow "gatto" \mid "ragazza" \mid "topo" \mid "libro"$
 $V \rightarrow "mangia" \mid "legge"$

2. Sistema di Riscrittura in DCG (Produzione Albero di Derivazione)

La grammatica in forma DCG, utilizzata per generare l'albero di derivazione in forma parentetica, è definita come segue:

```
% Grammatica in DCG
sentence(tree(s(NP, VP))) --> noun_phrase(NP), verb_phrase(VP
    ).

noun_phrase(tree(np(Det, N))) --> determiner(Det), noun(N).
verb_phrase(tree(vp(V, NP))) --> verb(V), noun_phrase(NP).

% Lessico
determiner(tree(det(il))) --> [il].
determiner(tree(det(la))) --> [la].
determiner(tree(det(un))) --> [un].
determiner(tree(det(una))) --> [una].

noun(tree(n(gatto))) --> [gatto].
noun(tree(n(ragazza))) --> [ragazza].
noun(tree(n(topo))) --> [topo].
noun(tree(n(libro))) --> [libro].

verb(tree(v(mangia))) --> [mangia].
verb(tree(v(legge))) --> [legge].

% Esempio di query:
% ?- phrase(sentence(Tree), [il, gatto, mangia, il, topo]).
% Tree = tree(s(tree(np(tree(det(il)), tree(n(gatto)))),
%               tree(vp(tree(v(mangia)), tree(np(tree(det(il)
%               ), tree(n(topo)))))))).
```

3. Sistema di Riscrittura per Forma Logica

Un ulteriore sistema DCG è progettato per tradurre le frasi in una rappresentazione logica basata su predicati verbali e forme nominali semplici:

```

% Forma logica
sentence(logical_form(VP)) --> noun_phrase(subject(S)),
    verb_phrase(predicate(VP, S)).

noun_phrase(subject(S)) --> determiner(_), noun(n(S)).
verb_phrase(predicate(Predicate, Subject)) --> verb(v(
    Predicate)), noun_phrase(object(O)),
    {Subject =.. [Subject], O =.. [O]}.

% Lessico
determiner(_) --> [il].
determiner(_) --> [la].
determiner(_) --> [un].
determiner(_) --> [una].

noun(n(gatto)) --> [gatto].
noun(n(ragazza)) --> [ragazza].
noun(n(topo)) --> [topo].
noun(n(libro)) --> [libro].

verb(v(mangia)) --> [mangia].
verb(v(legge)) --> [legge].

% Esempio di query:
% ?- phrase(sentence(logical_form(mangia(gatto, topo))), [il,
    gatto, mangia, il, topo]).
% Risultato: logical_form(mangia(gatto, topo)).

```

4. Esempio Completo

Consideriamo l'esempio:

- **Frase di Input:** *Il gatto mangia il topo.*

Produzione dell'Albero di Derivazione:

```

?- phrase(sentence(Tree), [il, gatto, mangia, il, topo]).
Tree = tree(s(tree(np(tree(det(il)), tree(n(gatto)))),
    tree(vp(tree(v(mangia)), tree(np(tree(det(il)),
    tree(n(topo))))))).

```

Produzione della Forma Logica:

```
?- phrase(sentence(logical_form(mangia(gatto, topo))), [il,  
    gatto, mangia, il, topo]).  
logical_form(mangia(gatto, topo)).
```