

Universita' degli studi di Pisa
Facolta' di scienze Matematiche, Fisiche e Naturali
Corso di Laurea in Informatica

Corso di Intelligenza Artificiale: Trattamento Automatico del Linguaggio Naturale

Anno accademico 2000/2001

Aspetti e problematiche del parser del linguaggio naturale

Elena Fiore

1. Il linguaggio naturale nella società dell'informazione

L'informazione scientifica, tecnica e relativa alle imprese raddoppia, oggi, in meno di un anno e tra il 70 e il 90% di tutta l'informazione digitale prodotta viene memorizzata in forma di testo nelle varie lingue naturali esistenti. Nello stesso tempo, crescono le possibilità di calcolo sia in termini di velocità che di massa di dati memorizzati.

La velocità e la massa, tuttavia, non rendono utili i sistemi se non vengono correlate alla possibilità di utilizzare selettivamente l'informazione in base a bisogni e scopi precisi. Tali scopi si realizzano attraverso la creazione e la reperibilità di contenuti che, nel caso di produzione in forma di lingua, sono significati legati a parole, a frasi o ad altre parti del discorso e del testo.

Dall'importanza della lingua come mezzo di comunicazione e di costruzione della conoscenza deriva la necessità della sua elaborazione elettronica per adeguarla ai nuovi supporti sui quali i suoi prodotti vengono generati, trasmessi e recuperati.

Le tecnologie avanzate del linguaggio umano sia scritto che parlato, permettono interscambi effettivi tra linguaggio e cultura, interfacce naturali per servizi digitali e per l'assimilazione e l'uso più intuitivi dei contenuti multimediali utilizzabili nei settori chiave come l'editoria aziendale e commerciale, l'istruzione e la formazione, i servizi pubblici e i gruppi con necessità particolari (disabili).

Occorre verificare quanta e quale conoscenza ed elaborazione linguistica di un testo sia necessaria per rendere più efficaci le procedure del suo recupero o per esprimere una domanda senza vincoli e con l'adeguata ricchezza sintattica e semantica. L'uso del linguaggio naturale per accedere ad informazioni contenute in basi di dati possiede indubbi vantaggi per il fatto di eliminare la necessità di conoscere il linguaggio formale richiesto dalla base di dati, di semplificare e rendere naturali alcune richieste altrimenti complesse, di permettere l'uso di espressioni anaforiche o ellittiche, senza dover, quindi, ripetere porzioni di richieste precedentemente eseguite.

Le descrizioni in linguaggio naturale dovranno poter variare sia come struttura, sia per i termini usati. Il linguaggio da usare dovrà poter essere telegrafico e sintetico ma, nello stesso tempo, dovranno essere accettate richieste parzialmente inappropriate o ridondanti. Affinché questi processi comunicativi divengano effettivi è necessaria un'elaborazione del contenuto dell'informazione per creare filtri per la sua produzione e la sua utilizzazione.

Tale elaborazione deve mirare, principalmente, alla realizzazione di:

- interfacce naturali per i servizi digitali, facili da usare tramite l'interattività naturale che siano in grado di accrescere la naturalezza dell'interazione uomo-macchina e l'efficacia della comunicazione interpersonale;
- di strumenti che migliorino l'efficacia dell'accesso all'informazione sfruttando la conoscenza linguistica incorporata in documenti, dialoghi, messaggi, basi di dati e audiovisivi.

Il raggiungimento di questi due obiettivi è subordinato alla realizzazione di strumenti quali: parser potenti e flessibili per il linguaggio naturale e indici per il recupero efficace di tutta l'informazione disponibile.

2. Che cos'è un parser e su cosa opera

Il parsing è il processo di costruzione di una rappresentazione lineare di un linguaggio in accordo con una data grammatica. Per ciascuna grammatica esistono generalmente un numero infinito di rappresentazioni lineari, chiamate frasi, che possono essere in essa costruite. Una data grammatica finita può, quindi, generare un numero infinito di frasi ed è questa la sua caratteristica più importante: essa riassume succintamente le strutture di un infinito numero di oggetti di una stessa classe.

Un parser è un programma che si articola in un numero ristretto di funzioni:

- scansione della frase in input;
- scansione della grammatica e reperimento delle regole rilevanti
- strutturazione della frase e memorizzazione progressiva (composizionale) di tale struttura.

La maggior parte dei parser vengono caratterizzati per quest'ultima funzione.

Esistono diverse ragioni per realizzare tale processo di strutturazione chiamato parsing.

La prima deriva dal fatto che le strutture da esso ottenute ci aiutano a processare gli oggetti con maggiore efficacia: infatti, una volta che le strutture di un documento vengono portate in superficie, possono essere manipolate più agevolmente.

Un secondo motivo è correlato al fatto che le grammatiche, in un certo senso, rappresentano il nostro modo di comprendere le frasi in esame. Il modo migliore per fornire una buona grammatica per un qualsiasi linguaggio è quello di comprendere, fino in fondo, tale linguaggio e le regole che governano le sue frasi.

Infine, un terza ragione risiede nella possibilità, propria di un parser, e soprattutto di un "error-repairing" parser, di provvedere al completamento dell'informazione mancante.

Data una grammatica ragionevole per un certo linguaggio, un "error-repairing" parser può suggerire le possibili classi di parole da inserire al posto di quelle mancanti o sconosciute.

Un linguaggio è un insieme infinito di possibili frasi correlate. Ciascuna frase consiste in una sequenza strutturata di parole che hanno un significato nel mondo reale. Struttura e significato danno alla frase il messaggio che essa comunica. Anche le parole, a loro volta, possiedono una struttura e sono composte da lettere; lettere che cooperano per attribuire un significato alle parole.

La grammatica del linguaggio è l'insieme di regole che permette di costruire le frasi legali in quello specifico linguaggio. Questo vuol dire che, seguendo pedissequamente queste regole, deve essere possibile costruire ogni frase del linguaggio e non le altre.

Tuttavia, data una frase, le regole non ci dicono come costruirla, ma solo che è possibile farlo.

Ma un linguaggio è, in primo luogo, un mezzo di comunicazione interpersonale da usare, per lo più, in maniera inconscia. È, infatti, il tramite di un processo di comunicazione nel quale sia l'emittente che il destinatario eseguono operazioni cognitive complesse.

L'emittente è motivato da scopi comunicativi e utilizza alcuni mezzi per essere compreso: egli deve tracciare un piano per concretizzare i suoi scopi in una sequenza di suoni o di segni grafici tracciabili su un qualche supporto stabile.

Da parte sua, il destinatario deve essere in grado di ripercorrere il processo inverso, deducendo gli scopi e i messaggi originali partendo dall'informazione ricevuta. In questo

processo di deduzione degli scopi comunicativi dell'emittente, un ruolo decisivo giocano alcuni aspetti quali la scelta delle parole, la struttura delle frasi e i modelli di enfasi e di intonazione. Il linguaggio naturale, infatti, è, per sua natura, sfumato ed ambiguo e una corretta interpretazione di ciò che viene percepito comporta, spesso, la risoluzione di problemi di ambiguità e il recupero, tramite processi di ragionamento, di informazione non esplicitamente comunicata. Nel processo di produzione linguistica, l'emittente, parte dai suoi scopi e coinvolge nel processo le conoscenze relative alla struttura della lingua che utilizza; la conoscenza del mondo necessaria a selezionare i concetti e a costruire così il significato da comunicare; le conoscenze relative alle modalità di realizzazione degli atti comunicativi.

Nel processo di ricezione, in modo inverso, il ricevente del messaggio, utilizza le proprie conoscenze per riuscire ad estrarre dal messaggio il significato in esso codificato.

Questo processo di codificazione/decodificazione implica, necessariamente, che le conoscenze utilizzate siano condivise tra gli interlocutori per garantire un' adeguata efficacia comunicativa.

Il linguaggio naturale ha, innegabilmente, una funzione ideativa che mette in relazione la capacità astratta degli esseri umani con la realtà e che determina, secondo alcuni linguisti, le forme attraverso le quali quella realtà viene compresa e mediata.

3. Elaborazione del linguaggio naturale

Nella comprensione del linguaggio naturale esistono diversi task da esaminare e trattare:

- analisi del segnale
- analisi sintattica
- analisi semantica
- pragmatica

Il problema dell'*analisi del segnale* si occupa dell'elaborazione e dell'estrazione di un segnale acustico e dell'individuazione delle parole in esso contenute e codificate.

L'*analisi sintattica* è relativa all'analisi della struttura delle frasi: frasi formate dalle stesse parole poste in ordine differente possono avere significato completamente differente.

L'*analisi semantica* si prefigge di assegnare un significato al risultato del parser esaminando il significato delle parole in esso contenute. Questa è la fase in cui possono essere risolte la maggior parte delle ambiguità.

La *pragmatica*, infine, interessa la comprensione delle frasi o di altro testo nel contesto della nostra base di conoscenza.

3.1 Analisi del segnale

Abbiamo già detto che l'*analisi del segnale* si occupa dell'elaborazione e dell'estrazione di un segnale acustico e dell'individuazione delle parole in esso contenute e codificate.

Infatti, per prima cosa, il segnale dovrà essere digitalizzato; in seguito, i vari frammenti, nel nostro caso i fonemi, andranno riconosciuti e assemblati per formare le parole.

Riuscire a ricostruire una parola partendo dai singoli fonemi risulta un problema delicato

dal momento che esistono diverse sequenze di fonemi che possono significare una cosa o un'altra in relazione al contesto nel quale sono inseriti.

Pertanto, occorrerà essere a conoscenza delle regole per la generazione o il riconoscimento di sequenze di fonemi che non violino il sistema formale di una particolare lingua, ovvero, essere in grado di svolgere un'accurata analisi lessicale che permetta di individuare ed, eventualmente, riprodurre, i token di un linguaggio, nel nostro caso le parole. Stiamo, difatti, analizzando un linguaggio al livello parole/lettere, livello che, dai linguisti viene, più propriamente, chiamato grammatica a sottolineare la distinzione rispetto al livello superiore frasi/parole che viene, invece, definito sintassi. Le regole relative al processo di derivazione permettono la formazione delle parole attraverso un insieme di trasformazioni. Questo processo permette di passare da parole di base a suffissati: a partire da alcune categorie di parole se ne possono ottenere di nuove applicando suffissi secondo determinate regole. Ad esempio, se si aggiungono a certi verbi alcuni tipi di suffissi si otterranno nomi la cui caratteristica semantica è quella di indicare un'azione o l'agente di un verbo. Oppure, esiste un tipo di suffissazione con la quale il significato della parola muta soltanto per alcuni aspetti (quantità, qualità, giudizio del parlante). Ma gli elementi strutturali di una parola possono essere molteplici: una radice che ne porta il significato base; un suffisso che, abbinato a tale radice, ne indica il grado (comparativo, superlativo, neutro); un suffisso flessionale che ne indica genere e numero. Tutte queste caratteristiche dovranno essere adeguatamente riconosciute e memorizzate in vista delle successive fasi di analisi sintattica e semantica.

Pertanto, nelle grammatiche orientate all'analisi del linguaggio naturale vengono aggiunte quelle che vengono chiamate **augmentations**, ovvero, delle ulteriori specificazioni alla grammatica che permettono di riconoscere i fonemi in accordo. L'inserzione di questi augmentations permette di tener adeguatamente conto delle restrizioni di cooccorrenza di origine morfosintattica permettendo, ad esempio, di verificare che ci sia accordo tra il soggetto e il verbo di una frase.

In una grammatica di questo tipo si presenta come segue:

(NP NUMBER ?n) → (ART NUMBER ?n) (N NUMBER ?n)

Questa regola ci dice che un NP è composto da due sub-costituenti, il primo un articolo, e il secondo un nome, in cui la augmentation NUMBER deve essere identica.

Per costruire una grammatica come questa è necessario che ogni componente sia caratterizzato nel lessico non solo in base alla categoria sintattica, ma anche in base ad altre caratteristiche morfosintattiche rilevanti. In questo modo è possibile introdurre e trattare adeguatamente quelle restrizioni esistenti nel linguaggio naturale che non potrebbero essere riconosciute da un semplice parser basato su una CFG.

3.2 Analisi sintattica e parsing

Assumiamo, ora, per semplicità, di avere a che fare con testo disponibile, direttamente, in forma scritta su un supporto stabile.

Un testo in forma scritta viene percepito e interpretato mediante il riconoscimento di segni grafici tracciati sulla carta. Esistono, perciò, in ogni lingua, ben definite regole che permettono di riprodurre il linguaggio su supporti stabili. La competenza sintattica non consiste solo nel saper riprodurre o riconoscere sequenze ben formate data la grammatica di una lingua, ma, anche, nel saper individuare i componenti di queste sequenze e i loro

rapporti strutturali e funzionali. Può accadere, ad esempio, che lo stesso insieme di parole con due ordini differenti possa voler dire cose completamente differenti. Perciò, la nostra conoscenza non si può limitare, esclusivamente, a richiamare alla memoria le parole che abbiamo memorizzato e ad assemblarle in frasi secondo regole e strutture precise, ma si deve occupare, soprattutto, di riconoscere gli elementi di cui tali frasi sono formate. Il meccanismo che seleziona le strutture e definisce le regole per la loro elaborazione fa parte delle nostre facoltà mentali.

Attraverso la fase di analisi del segnale, siamo in grado di individuare (o produrre) sequenze ben formate di suoni o di caratteri, le parti di cui sono composte e il loro valore funzionale e semantico. Possiamo, perciò, riconoscere il loro significato e categorizzarlo secondo determinati parametri. Questa categorizzazione semantica sarà utilizzata, successivamente, al livello frasi/parole per verificarne la correttezza strutturale. Un modo semplice di classificazione semantica si basa su distinzioni come quelle tra oggetti animati e inanimati o tra animati maschi e femmine. Questa classificazione è richiesta, ad esempio, per l'uso corretto dei pronomi o degli aggettivi. Le caratteristiche semantiche delle parole vengono utilizzate anche per costruire frasi corrette in cui venga rispettata la compatibilità semantica tra il verbo e, rispettivamente, il soggetto e il complemento oggetto. Soggetto e verbo devono concordare in persona e numero. Quello che vogliamo è che venga riconosciuta come legale una frase del tipo "io parlo" e che non ci sia, invece, un parser per una frase quale "io parla". Deve esistere, pertanto, un'intersezione non nulla tra il numero e la persona del soggetto e i possibili numero e persona del verbo. Prendendo in considerazione, ad esempio, il congiuntivo "parli", esso ha come numero il singolare, ma ha, come possibili persone, l'insieme $\{1^\circ, 2^\circ, 3^\circ\}$. Il soggetto di tale verbo dovrà, quindi, essere in numero singolare ma potrà essere in qualsiasi persona.

Ogni lingua ha costruito, inoltre, alcuni meccanismi per indicare se gli oggetti cui ci riferiamo siano conosciuti dal parlante e/o dall'ascoltatore. Esistono, pertanto, alcuni elementi linguistici che ci permettono di indicare ai nostri interlocutori alcune entità, siano esse concrete o astratte, appartenenti al mondo reale o costruite in un qualche universo concettuale di riferimento sulle quali intendiamo comunicare qualcosa. La funzione degli elementi identificatori non è, tuttavia, sempre univoca: ciascuno può svolgere più funzioni differenti di individuazione e può generare problemi di ambiguità la cui risoluzione è, spesso, subordinata all'esistenza di una base di conoscenza sufficientemente informata.

Ogni fruitore di una lingua è a conoscenza del passato, del presente e del futuro e può descriverne le relazioni. La scelta dei tempi del verbo riflette alcune correlazioni temporali che percepiamo nello svolgersi degli eventi reali e porta, quindi, con sé informazione semantica da considerare in fase di analisi.

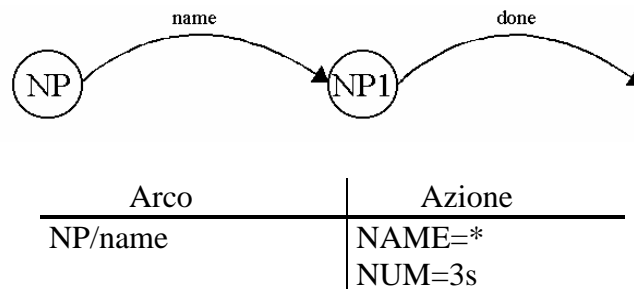
Molti significati della lingua parlata sono comunicati dalla presenza dell'intonazione o dell'accentazione. Queste non sono riflesse, direttamente, nella scrittura benchè la presenza della punteggiatura fornisca informazioni ad esse correlate. La prosodia ci permette, così, di segmentare correttamente la sequenza fonica riconoscendo i componenti di una frase o del discorso. Essa ha, anche, un valore aggiuntivo, permettendoci di comunicare altri significati oltre a quello base veicolato dall'ordine delle parole.

Il linguaggio naturale ha, innegabilmente, una funzione ideativa che mette in relazione la capacità astratta degli esseri umani con la realtà e che determina, secondo alcuni linguisti, le forme attraverso le quali quella realtà viene compresa e mediata.

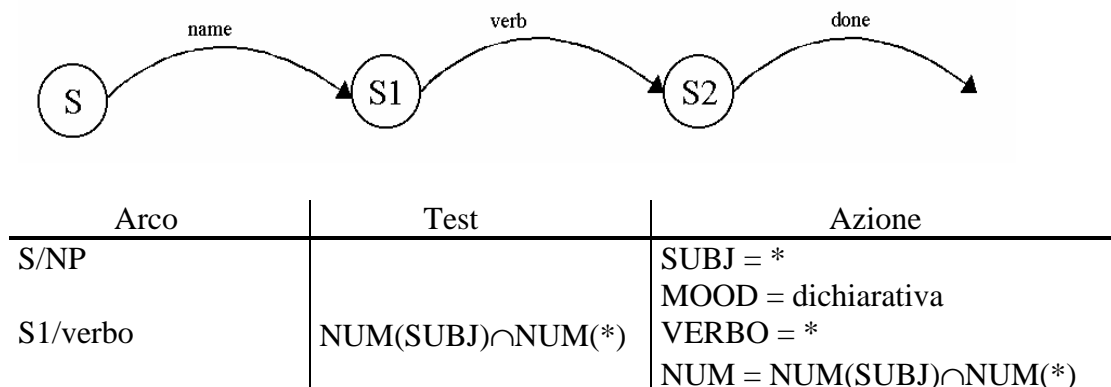
Prendiamo ora in considerazione come possibili implementazioni di un parser quelle che fanno uso dei cosiddetti Augmented Transition Networks o ATN e dei Chart Diagram.

3.2.1 ATN

Un ATN è un grafo nel quale i nodi sono etichettati con nomi di altri puntatori di tipo ATN, mentre gli archi sono contrassegnati da categorie di parole o da identificatori (puntatori) di altri ATN. Si entra in un ATN quando si incontra un nodo di tipo NP (noun phrase) e si esce quando si attraversa un arco di tipo “done”. Durante l’attraversamento degli archi l’ATN assegna una serie di valori ai vari slots o augmentations delle strutture che, di volta in volta, vengono create. Per attraversare un arco etichettato con l’identificatore di un altro ATN, deve essere invocata la network corrispondente all’etichetta ed il risultato da essa restituito sarà utilizzato nel proseguimento dell’analisi. Quando si incontra un arco di tipo “done” si esce dall’ATN e si restituisce una lista individuata dal nome del nodo radice dell’ATN. Ciascun elemento della lista è una coppia in cui il primo elemento è uno slot inserito dall’ATN (ad esempio NAME, NUMBER, etc.), mentre il secondo elemento è il suo valore. Esempio molto semplici di ATN sono i seguenti in cui viene eseguito il parser del nome “Mary”:



e della frase “Mary cammina” con relativo test di correttezza:



Ovviamente, nel momento in cui si fa riferimento alle categorie di parole si deve presupporre l’esistenza di un dizionario che associ, ad ogni entry, la corrispondente categoria sintattica ed altre informazioni da utilizzare durante questa fase di analisi sintattica.

Un arco potrà essere caratterizzato, oltre che da una categoria lessicale o da un identificatore di ATN, anche, dalla presenza di un test. Tale test indica che esso può essere attraversato solo se il test viene superato. Per percorrere un arco, quindi, non è necessario, soltanto, che la parola che viene processata sia dell'appropriata categoria sintattica, ma, anche, che il test ad essa applicato, qualora ci sia, dia esito positivo. Un altro caso di arco speciale è l'arco di tipo "jump". Infatti, in alcuni casi, i componenti grammaticali di una frase hanno dei sottocampi opzionali, come nel caso dei verbi composti per i quali gli ausiliari risultano opzionali. In questo caso occorrerà controllare che il costrutto verbale sia corretto e che l'ausiliare (essere o avere) sia quello appropriato. Una volta fatto questo, i campi opzionali saranno ignorati e l'unica informazione memorizzata sarà relativa, esclusivamente, al verbo principale e al suo tempo (passato prossimo, trapassato prossimo, etc.).

Per eseguire il parser abbiamo bisogno di tener traccia di due puntatori: uno alla nostra locazione corrente nell'ATN e l'altro alla locazione che stiamo esaminando nella frase. Il puntatore di tipo ATN è una lista di puntatori: quando saltiamo da un ATN ad un altro, infatti, dobbiamo memorizzare il punto di ritorno nell'ATN originale.

Inizialmente, il puntatore ATN ha il valore (S); questo vuol dire che siamo all'inizio dell'ATN che analizza completamente le frasi.

In sintesi, per analizzare una frase, dovremo eseguire il seguente algoritmo:

1. Settare il puntatore ATN ad (S) e il puntatore alla locazione nella frase all'inizio della frase in questione.
2. Scegliere un arco di uscita dal nodo corrente. Per poter attraversare "legalmente" l'arco dovranno essere verificate le seguenti condizioni:
 - (a) Tutti i test dovranno essere soddisfatti dal valore della variabile corrente.
 - (b) Se l'arco è etichettato con una categoria di parole, la parola corrente dovrà appartenere a tale categoria.
3. Eseguire le azioni associate all'arco. In più, in dipendenza dal tipo di arco, eseguire le seguenti azioni:
 - (a) Se l'arco è una categoria di parole, aggiornare, di una parola, la posizione corrente nella frase e sostituire il nodo corrente con il destinatario dell'arco in esame;
 - (b) Se l'arco corrisponde ad un altro ATN, inserire il nodo iniziale dell'ATN nella pila dei puntatori ATN;
 - (c) Se l'arco è di tipo jump, sostituire il nodo corrente con il destinatario dell'arco in esame;
 - (d) Se l'arco è di tipo done, eliminare il nodo corrente dai puntatori ATN e settare * al valore restituito da questo nodo. Se il puntatore ATN è ora vuoto e tutte le parole del testo sono state processate, restituire *. Se il puntatore ATN è vuoto e rimane testo da analizzare, fallire. Altrimenti ritornare al passo (2).

Nei passi dell'algoritmo il simbolo * indica lo stato corrente dell'ATN.

L'approccio ATN è di tipo bottom-up.

Consideriamo ora le "prepositional phrase nouns".

Queste strutture possono modificare sia frasi nome che l'intera frase cui si riferiscono. Possono essere costituite, oltre che da una frase nome, anche da una singola parola come accade, ad esempio, con gli avverbi di tempo. Questo tipo di strutture viene trattato in modo del tutto analogo alle precedenti utilizzando sempre ATN resi più complessi dall'aggiunta di un nodo.

Inoltre, avremo bisogno di aggiungere uno slot “modificatore” alla frase o alla frase nome. L’azione da compiere quando si incontra una frase preposizionale è:

MODS=append(mods,*).

Questo aggiunge la frase preposizionale alla lista di quelle modificate. In questo stadio dell’analisi rimane, tuttavia, un’ambiguità relativa alla risoluzione del suo riferimento; tale ambiguità sarà risolta nella successiva fase di analisi semantica.

Infine, andiamo ad esaminare le frasi interrogative.

Le frasi interrogative che andremo a considerare sono quelle che iniziano con “chi” o con “cosa” (ma è analogo il procedimento nel caso degli altri pronomi o avverbi interrogativi). L’idea di base è la seguente: quando incontriamo un “chi” o un “cosa” all’inizio di frase facciamo due cose:

1. primo, cambiamo il modo della frase da dichiarativa in interrogativa.
2. secondo, realizziamo che il “chi” o il “cosa” prende il posto di una phrase nouns situata in un qualche punto della frase; perciò, memorizziamo tale pronome come un frase nome virtuale da usare più tardi.

Per eseguire quest’ultimo punto, settiamo il valore di una speciale variabile, HOLD, alla frase nome “chi” o “cosa”. Quando usiamo la frase nome virtuale, settiamo HOLD a nil per indicare ciò che abbiamo fatto in questo modo. Noi non permettiamo che l’arco done venga attraversato senza che l’hold NP sia stato realmente usato. Questo viene fatto per garantire che non possa esistere un parser per frasi come “Cosa Mery da ‘ all’uomo dei suoi sogni una foto?”.

3.2.2 Chart Diagram

Abbiamo già detto come l’approccio utilizzato nella costruzione di un parser tramite ATN sia di tipo bottom-up.

Nella fase di scansione della grammatica e di reperimento delle regole rilevanti, un parser utilizza in genere due strategie. Partendo dalla regola più alta si procede a riscrivere le regole sempre più basse fino ad incontrare la regola i cui simboli a destra della freccia soddisfano la stringa in input: questa tecnica è detta top-down. Oppure, partendo dal simbolo nella frase, si cerca la o le regole la cui parte destra soddisfi la stringa in input: questa tecnica è detta bottom-up.

La strategia top-down è altamente predittiva, visto che utilizza regole di riscrittura da sinistra verso destra, indicando già la possibile struttura della frase. In questo modo, è ridotta l’ambiguità delle parole che, considerate all’interno di una frase, vedono limitata la possibilità di ricoprire taluni ruoli sintattici. La strategia bottom-up non possiede, invece, questo vantaggio dato che espande i nuovi archi tenendo conto solo del primo costituente riconosciuto.

D’altra parte, ogni volta che l’algoritmo top-down ha bisogno di fare backtracking esso torna ad esaminare nuovamente i componenti della frase che aveva precedentemente processato per predizioni che si sono rivelate errate. L’algoritmo bottom-up, invece, esamina ogni elemento di una frase solo una volta evitando di riscriverlo più volte se utilizzato in diverse regole.

Vista la situazione, è possibile, combinando i due metodi, tentare di avvantaggiarsi delle peculiarità di uno e dell’altro. La strategia ottenuta in questo modo sarà altamente predittiva, come la top-down, ed eviterà duplicazioni di lavoro, come la bottom-up. Tale metodo è detto

Chart Parsing.

L’idea alla base di questo parser è che una volta che una struttura è stata trovata, essa può essere riutilizzata altrove anche se il costituente per il quale era stata proposta come componente si è rivelato essere errato. Molti parser sono dotati di meccanismi di questo genere chiamati *tavole*

delle stringhe ben-formate o **charts** i quali tengono traccia dei costituenti che sono stati espansi durante parte del parser ma che possono essere usati in altre regole.

Un chart può essere visualizzato come un grafo i cui nodi rappresentano punti nella frase e i cui archi rappresentano, invece, i costituenti. Un arco prende il nome del costituente che inizia e finisce ai nodi che connette. Il chart iniziale contiene solo gli archi corrispondenti alle singole parole dell'input e alle loro categorie lessicali. Non appena viene individuato un costituente, esso viene aggiunto al chart.

Il parser si serve del chart introducendovi un nuovo costituente ogni volta che una regola viene applicata con successo e cercando al suo interno per altri costituenti ogni volta che c'è un non-terminale all'inizio della remainder. Se c'è un arco per il corrispondente simbolo nel punto corrente nel chart, il parser può avanzare nella sentenza fino al punto di fine di quell'arco senza bisogno di eseguire nuovamente l'analisi del costituente. Questo può portare al risparmio di una notevole quantità di inefficienza combinatoriale rispetto al parser top-down.

In realtà, all'interno di un chart deve essere memorizzata molto più che l'informazione relativa alle categorie lessicali delle parole della frase di input e al nome dei costituenti. Infatti, se un costituente ha successo con un particolare parser, ma dei problemi si sviluppano in seguito, deve esistere un modo per conoscere quale regola è stata tentata e quale no.

Ogni nodo corrisponde allo spazio tra due parole (più l'inizio e la fine); perciò, ci sarà un numero di nodi pari al numero di parole della frase di input aumentato di 1.

Gli archi di un chart possono essere completi o attivi. Un arco si dirà completo quando la sua remainder è vuota, altrimenti si dirà attivo.

Un arco completo rappresenta un costituente che è stato individuato. Un arco attivo rappresenta un punto lungo la via della ricerca di un costituente, la sua remainder che cosa deve ancora essere trovato. Un arco può essere specificato tramite il numero del nodo di inizio e del nodo di fine insieme con la rappresentazione della regola della grammatica su cui è basato e la remainder, ancora da completare, di quella regola. Il numero del nodo di fine viene utilizzato per separare la parte della regola che è già stata analizzata dalla remainder.

Perciò una regola del tipo:

$({}_4\text{NP} \rightarrow \text{DET } {}_5\text{NP}_2)$

rappresenta un arco attivo con nodo di inizio 4, nodo di fine 5, che sta analizzando un costituente di tipo NP, che ha già trovato il suo DET e che ha ancora bisogno di NP2 per diventare completo. Un arco completo avrà il numero di nodo di fine alla fine della regola come in: $({}_4\text{NP} \rightarrow \text{DET NP}_2 {}_6)$

Il processo di riconoscimento di una stringa di input attraverso un chart è, sostanzialmente, un ciclo nel quale, ad ogni iterazione, un nuovo arco viene introdotto nel chart. Come conseguenza di questo inserimento, c'è una combinazione degli archi appena introdotti insieme con altri preesistenti nel chart e la proposta di altri archi potenziali con cui gli ultimi archi introdotti potrebbero combinarsi. Ad ogni iterazione c'è una lista di archi pendenti che aspettano di essere aggiunti al chart. L'inizializzazione del chart inserisce alcuni archi iniziali in tale lista e sia l'attività di ricerca che quella di proposta possono inserirne di nuovi. L'attività di "combinazione" degli archi è bottom-up poiché essa tratta solo con archi già presenti nel chart. L'attività di "proposta" è invece top-down portando ad una ricerca di nuovi archi basandosi sul fatto che essi potrebbero essere usati per il match del remainder di alcuni archi attivi.

Combinando le due direzioni in questo modo e tenendo traccia di qualsiasi cosa nel chart, possiamo evitare le inefficienze dei due metodi applicati singolarmente.

Perciò, l'algoritmo di riconoscimento di una frase tramite chart può essere riassunto come segue:

- il chart viene inizializzato in modo da rappresentare le parole della frase di input con le relative categorie lessicali e propone il simbolo iniziale della grammatica come primo nodo del chart. In questo modo, nella lista degli archi pendenti ne sarà presente almeno uno.
- ripete i seguenti passi:
 - rimuove ciascun membro della lista degli archi pendenti del chart e lo considera come nuovo arco;
 - combina il nuovo arco con il chart (questo potrebbe produrre nuovi archi pendenti);
 - se il nuovo arco è attivo propone il primo simbolo del suo remainder come suo nodo finale nel chart (questo può produrre nuovi archi pendenti)

Il parser riesce se, ad un certo punto, tra gli archi o tra gli archi pendenti ce n'è uno completo tale che:

- il suo nodo di inizio è uguale al primo elemento dei nodi del chart
- il suo nodo di fine è uguale all'ultimo elemento dei nodi del chart
- la sua label è uguale al simbolo iniziale della grammatica

Se non ci sono archi pendenti quando deve esserne scelto uno, il parser fallisce.

Nell'algoritmo precedente non abbiamo specificato la strategia da seguire nella scelta degli archi pendenti all'interno della lista. Questo sostanzialmente perché la scelta della strategia non influenza le virtù computazionali di un chart parse. La scelta della strategia, infatti, può essere usata per modificare il comportamento dettagliato del processo. Ad esempio, stabiliamo che l'algoritmo debba tentare di finire un intero parser basato sulla regola scelta di volta in volta prima di esplorare le alternative. Se a ciò aggiungiamo una regola di stop che termina il parser non appena trova un arco completo per il simbolo iniziale della grammatica dall'inizio alla fine della sentenza, il parser avrà alcune proprietà del backtracking parser. Alterando la strategia ed eliminando la regola di stop il processo potrà, invece, assumere le caratteristiche di una breadth-first o di una depth-first e così via.

3.3 Analisi semantica

L'analisi semantica mira, facendo uso del significato delle parole, ad estendere e, forse, disambiguare il risultato del parser sintattico. Infatti, la struttura di una frase non può rappresentare da sola il suo significato.

La semantica permette di analizzare un'espressione del linguaggio naturale e di riconoscere le diverse interpretazioni che essa può assumere; questo è fatto utilizzando un linguaggio del primo ordine.

Questo linguaggio è utilizzato per codificare i significati delle parole e le loro possibili relazioni semantiche all'interno di una frase. Utilizzare un parser, accoppiato con un analizzatore semantico, consente di ridurre le possibili interpretazioni di una frase, così da snellire notevolmente il lavoro per il suo riconoscimento.

Questa nuova analisi è basata ancora una volta su regole, o meglio, su insiemi di regole. Un insieme di regole può tradurre in forma non ambigua un'intera frase, un altro le frasi nome e così via. Esisteranno, ovviamente un gran numero di regole, prendiamone in considerazione una per chiarezza:

Componente	Trigger	Condizione	Asserzione
------------	---------	------------	------------

NP	(MODS (PP (PREP in) (OBJ x)))	Time(*) \wedge time(x) Oggetto-fisico(*) \wedge locazione (x)	Durante(*,x) Locazione(*,x)
----	---------------------------------	--	--------------------------------

Siamo in presenza di una regola ambigua. Consideriamo, infatti, le due asserzioni: "un giorno in giugno" e "l'uomo in casa". Se il nome principale e la frase nome oggetto dell' "in" sono entrambi tempi, il significato di "in" è "durante". Se, invece, il nome principale è un oggetto fisico e la frase nome è un luogo, "in" specifica la locazione di un oggetto. Ovviamente, non è possibile mischiare le due accezioni poiché si arriverebbe a frasi senza alcun significato. La particolarità di questa regola è che per essa esistono due potenziali risultati che dipendono dall'eventuale significato di "in".

Resta, ora, da stabilire la correlazione e l'ordine di esecuzione tra analisi sintattica ed analisi semantica di un testo.

Il modo più semplice di procedere potrebbe essere quello di cominciare l'analisi semantica nel momento in cui abbiamo completato quella sintattica. Oppure, potremmo anche "accendere" le varie regole sintattiche non appena il procedere dell'analisi sintattica ne renda disponibili gli elementi necessari. Questo modo di procedere ha il vantaggio di eliminare immediatamente alcuni parser ambigui, utilizzando considerazioni di carattere semantico. Inoltre, esso risulta maggiormente affine a quello che succede nei processi umani di comprensione del linguaggio, processi nei quali la comprensione sintattica e semantica procedono in parallelo. Infatti, la gente, normalmente, non ha problemi a comprendere frasi grammaticalmente scorrette quando discorre, proprio perché, attraverso un'inconscia analisi semantica, riesce a intendere in modo univoco le asserzioni che ascolta.

3.4 Pragmatica

In questa fase si ricorre alla conoscenza nel nostro dominio per eliminare le ambiguità residue nella comprensione delle frasi e per interpretare queste frasi in un contesto più ampio. Tra le ambiguità che rimangono da esaminare ci sono quelle relative alla risoluzione dei riferimenti dei pronomi ovvero quella che comunemente viene definita risoluzione dell'anafora. La soluzione più immediata a questo problema potrebbe essere quella di individuare il più recente tra i possibili referenti possibili ed prendere in considerazione quello. Per far questo occorrerà utilizzare, naturalmente, informazioni sul dominio e spesso questo risulta ancora insufficiente per superare l'ambiguità. Un altro fattore che caratterizza il processo di comprensione del linguaggio attuato dall'uomo è l'analisi delle intenzioni dell'agente che parla o di cui si sta discutendo. Da un punto di vista formale, questo processo di ricognizione dei piani non è tanto complicato da capire. Noi abbiamo un agente A ed abbiamo bisogno di individuare l'obiettivo che A potrebbe avere e quali sono le azioni di tale agente A che fanno parte del suo piano per ottenere il suo obiettivo. L'assunzione fondamentale in questo caso è che la gente non agisce in modo casuale ma sempre perseguendo, più o meno razionalmente, degli obiettivi.

È possibile definire due forme di conoscenza del dominio: conoscenza del mondo e conoscenza della situazione.

- Conoscenza del mondo: si intende la conoscenza di restrizioni generali relative al mondo ed alle definizioni semantiche delle parole della lingua.

- Conoscenza della situazione: si riferisce alla conoscenza specifica della situazione, importante in molti casi, soprattutto per determinare il referente di una frase e per disambiguare le parole di una frase in base a ciò che “fa senso” in quel contesto.

4. Conclusioni

Le problematiche trattate finora sono mirate ad esplicitare l'intrinseca complessità strutturale di un processo inconscio ed istintivo per l'uomo quale la comunicazione attraverso il linguaggio naturale. D'altra parte, proprio il livello inconscio che caratterizza lo svolgersi di questo processo determina, nello stesso tempo, l'importanza della sua trattazione automatica al fine di facilitare e rendere immediato l'utilizzo dei servizi informatici di alta diffusione. La prospettiva di una interazione uomo-macchina in linguaggio naturale e la possibilità di un accesso efficiente e flessibile a tutta l'informazione esistente, rendono il campo della linguistica computazionale uno dei più interessanti e ricchi di sviluppi.