Università degli Studi di Siena
Facoltà di Lettere e Filosofia
Corso di Laurea in Scienze della Comunicazione
Esame di Linguistica computazionale
A.A. 2002/2003

Sistemi di Riconoscimento Vocale

Tesina di Claudio Lodoli

Cosa sono gli SRV?

I sistemi di riconoscimento vocale (SRV) consentono il controllo di un computer attraverso comandi vocali.

La parte software è un'applicazione che elabora gli input dell'utente e produce un risultato, che può avere forme diverse.

L'hardware necessario è costituito da un processore di media potenza (Pentium III con 128 MB di RAM) dotato di una scheda audio e di un microfono.

Breve Storia del Riconoscimento Vocale

La prima “macchina parlante” fu costruita nel 1769 da Von Kampelen ed era costituita da una scatola contenente… un uomo! Il primo reale tentativo di creare una macchina per il riconoscimento vocale risale, invece, alla seconda metà dell’Ottocento: Alexander Bell cercò di costruire un apparecchio che potesse aiutare i non udenti, trascrivendo ciò che veniva detto dagli altri.

Durante la Seconda guerra Mondiale, si ottennero i primi risultati nella sintesi vocale, ma lo sviluppo del RV andò a rilento fino agli anni Settanta. Solo negli anni Novanta, però, si sono avuti risultati soddisfacenti e solo negli ultimi anni, grazie alla potenza di calcolo raggiunta dai computer, la tecnologia per il RV si è potuta diffondere capillarmente.

Un particolare da sottolineare è che, a livello linguistico, il problema del RV è rimasto lo stesso da molti anni a questa parte: il vero ostacolo allo sviluppo dei SRV era l’insufficiente potenza dei computer.

Alcuni termini

Enunciato: una qualsiasi cosa detta dall'utente e compresa tra due momenti di silenzio.

Pronuncia: come il SRV si aspetta che una parola venga pronunciata.

Grammatica: ciò che il SRV è in grado di riconoscere, il contesto in cui lavora.

Vocabolario: le parole che il SRV riesce a comprendere

Training: fase di addestramento del SRV, durante la quale il sistema memorizza la pronuncia di un determinato speaker.

Accuratezza: la misura dell'abilità del SRV

Tratta da P. Nenad, "Natural Language Processing andSpeech Enabled Applications" - University of Sheffield

Tipologie di SRV

Speaker dependent: sono i sistemi che dipendono in modo determinante dall'utente. Il funzionamento è ottimale solo se vengono usati dall'utente che li ha "addestrati".

Speaker independent: il funzionamento è ottimale con qualsiasi utente.

La maggior parte degli SRV odierni appartiene al secondo gruppo, ma per molte applicazioni gli SRV del primo, più semplici e meno costosi, sono ancora all'altezza del compito richiesto.

Gli SRV ed il parlato

A seconda del loro utilizzo, gli SRV possono riconoscere:

Parole isolate: chi parla deve pronunciare una sola parola alla volta, molto spesso suggerita dallo stesso SRV (telefonia)

Sequenze di parole: l'utente può pronunciare una sequenza di parole senza la necessità di interrompersi (affari, medicina).

Parlato naturale: il sistema è in grado di riconoscere e processare quello che viene detto durante una qualsiasi conversazioni, riuscendo a individuare anche espressioni gergali, intercalari…

Ovviamente, la complessità del software e la potenza necessaria aumentano passando dal primo all'ultimo tipo.

Come funziona un SRV?

Fase 1: La macchina riceve un segnale vocale e lo riconosce come tale.

Fase 2: Elabora il segnale trasformandolo in una stringa di bit analizzabile.

Fase 3: Cerca una corrispondenza tra il segnale ricevuto e quelli che ha in memoria.

Fase 4: Restituisce un risultato, sia in caso positivo che negativo

Fase 1: ricezione e riconoscimento

Attraverso un microfono, il SRV riceve l'input dall'utente.

Il primo problema che la macchina deve affrontare è riconoscere l'input: deve discernere tra il vero input e il rumore che può provenire dall'ambiente.

Si può ridurre il rumore esterno usando microfoni unidirezionali, ma molto dipende sia dall'ambiente che dall'utente stesso, dal suo modo di parlare, dalla cadenza delle parole, dalle pause…

Fase 2: l'elaborazione

L'input vocale, per poter essere elaborato ed ottenere risposta, deve essere trasformato dal formato analogico al formato digitale: il suono diviene una stringa di bit.

L'utente parla, il computer "ascolta", …

…elabora l'input ricevuto…

1100101001010100000111

1100101001110000000111

1100001010101000000111

1100100011110100000111

1100101001010101110000

… e lo trasforma in una stringa binaria.

Fase 3: pattern matching

La stringa di bit ottenuta nella fase 2 viene confrontata con i modelli acustici presenti in memoria.

L'elaboratore sfrutta un modello statistico della distribuzione degli eventi acustici, in sostanza approssima l'input ricevuto e cerca il modello acustico che assomiglia di più all'evento.

Il modello statistico si basa sulla seguente formula:

(1) $$p(w/x)=\frac{p(x/w)p(w)}{p(x)}$$

p(w/x) è la probabilità che l'evento sia una certa parola.

p(x/w) è il modello acustico.

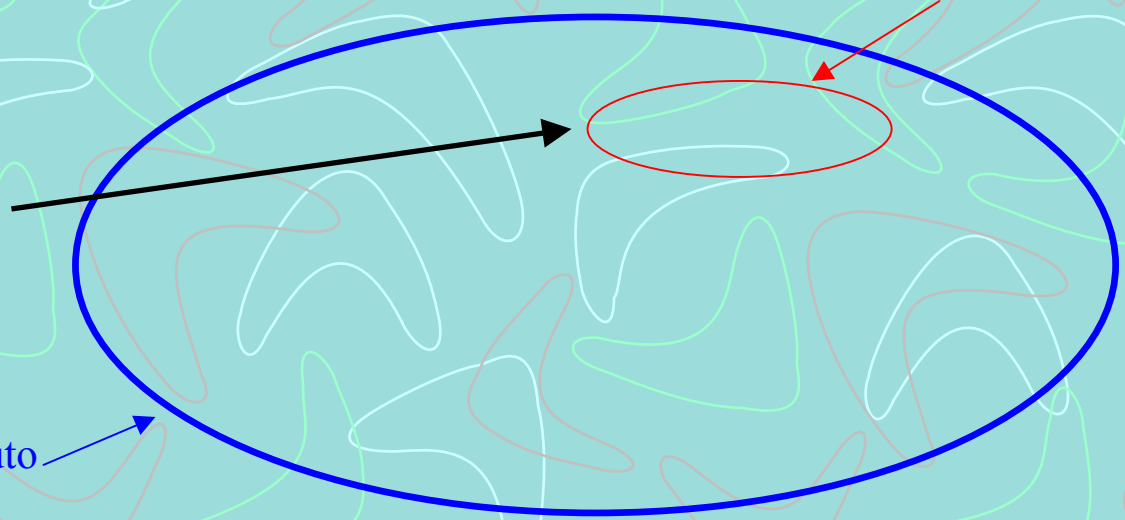
p(w) è il modello del linguaggio.

p(x) è l'evento.

Fase 3: pattern matching (2)

La formula (1) è un calcolo di probabilità basato sulla “storia” degli eventi precedenti, rappresentati da p(x) e p(w), combinati con il modello acustico del SRV. Da essa, il sistema ricava la probabilità che un enunciato corrisponda ad una certa parola, riuscendo a restringere il numero di eventi memorizzati con cui confrontare la produzione dell’utente.

$$p(w/x) = \frac{p(x/w)p(w)}{p(x)}$$



Fase 4: I risultati

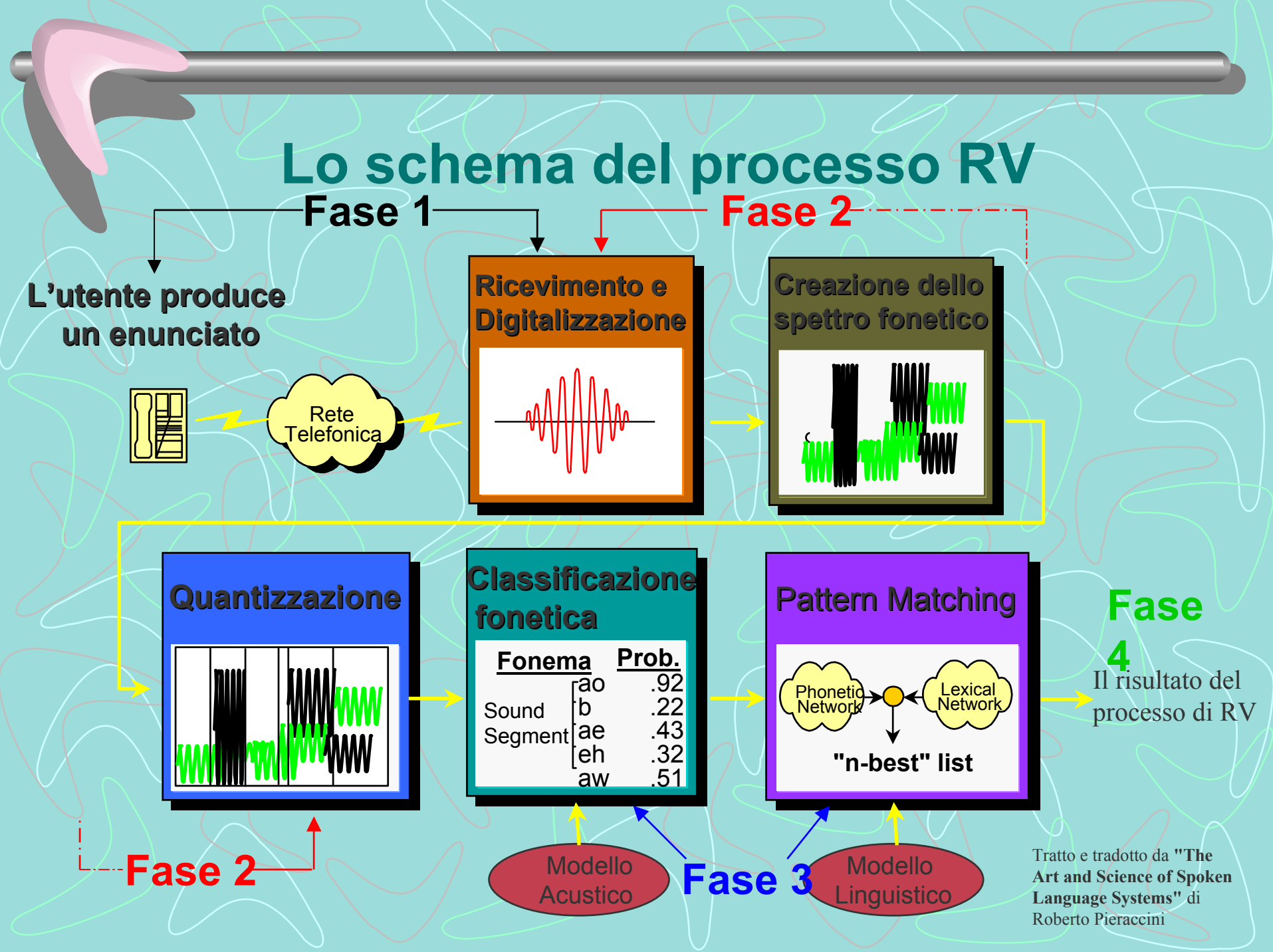
Il risultato del processo di RV può avere diverse forme:

Testo: nel caso di un SRV che serve a dettare lettere o documenti o che sia di aiuto a persone non udenti.

Sintesi vocale: nel caso di risponditori automatici per i servizi di informazione, o di computer “intelligenti” (vi ricordate HAL 9000 di “2001 Odissea nello spazio”?).

In caso di non riconoscimento, il SRV darà come output un messaggio d’errore o la richiesta di ripetere il comando o la parola.

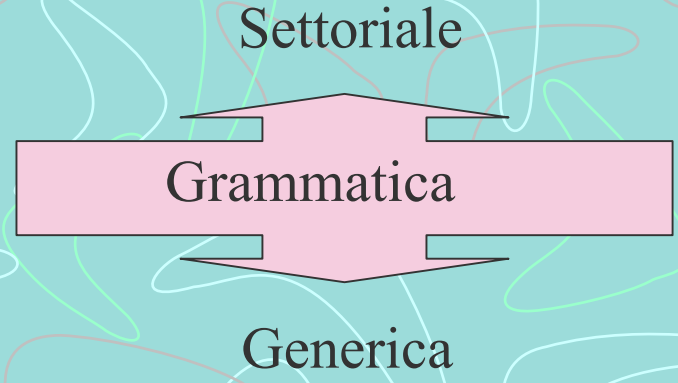
Lo schema del processo RV



Tratto e tradotto da **"The Art and Science of Spoken Language Systems"** di Roberto Pieraccini

Alla base del sistema

La base di un SRV è la sua grammatica: in essa vengono specificate tutte le parole e le espressioni che il sistema sarà in grado di riconoscere. Costituisce il contesto di lavoro del SRV.



La scelta del programmatore dipende dall'uso che verrà fatto del SRV.

Grammatica generica

I SRV che usano grammatiche generiche sono destinati, nella maggior parte dei casi, ad una utenza con esigenze di supporto per la creazione di documenti o a risponditori automatici.

Una grammatica generica è generalmente molto estesa: può contenere parole, frasi, modi di dire, regole.

L'accuratezza di un SRV basato su una G.G. può risultare minore rispetto a quella di uno basato su una grammatica settoriale, avendo più possibilità di confronto.

Grammatica settoriale

E’ destinata a macchine che operano in ambiti specifici, in cui solo certe produzioni accadono con una certa probabilità.

Può essere impiegata nei laboratori, nell’industria, in risponditori automatici dedicati ad un determinato servizio, com’è ad esempio il caso del risponditore delle Ferrovie Italiane.

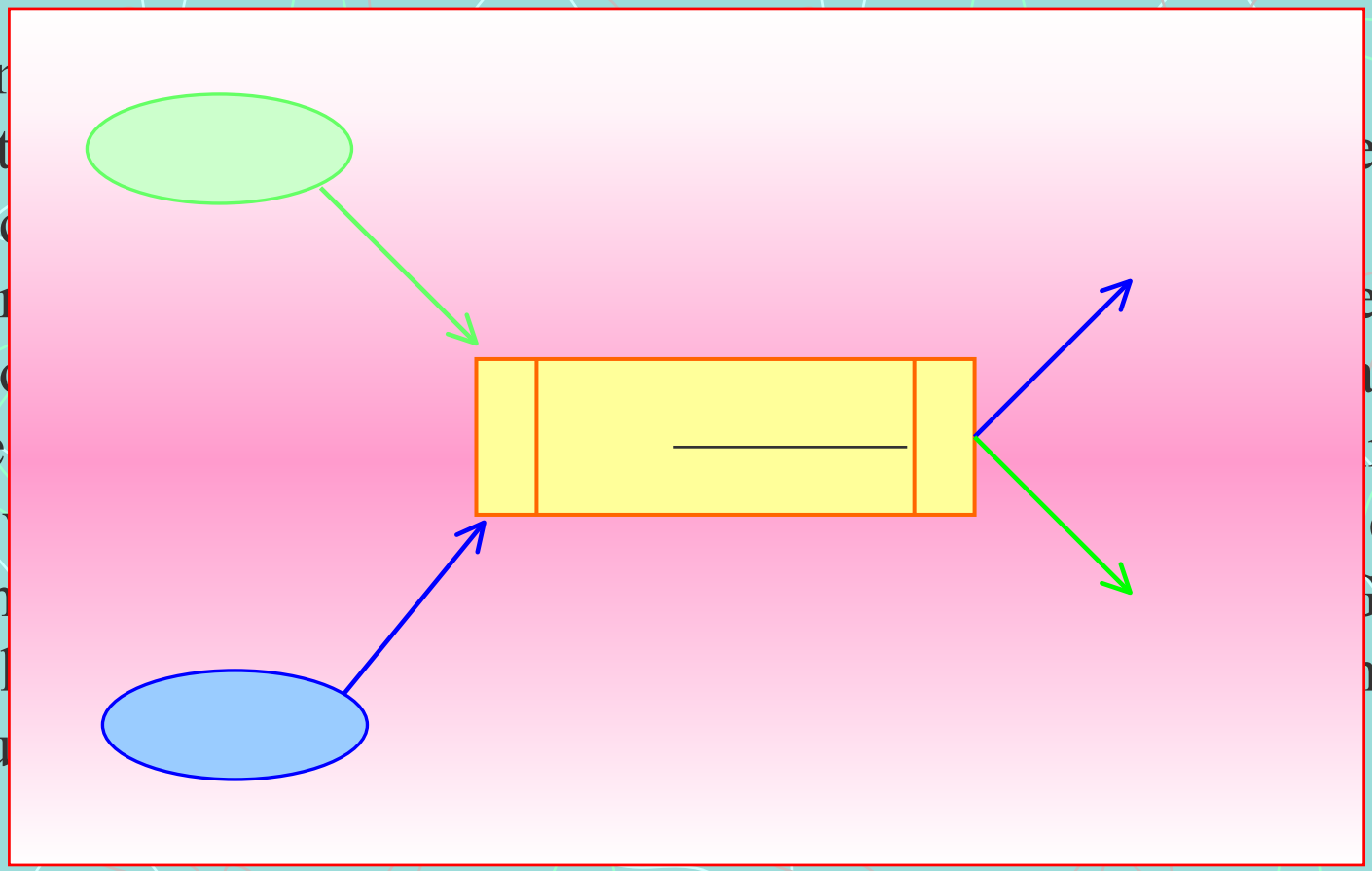
Generalmente, tendono ad accrescere l’accuratezza del SRV, limitando il numero di eventi possibili.

Un esempio di grammatica

Il seguente esempio, tratto da **Speech Recognition Grammar Specification Version 1.0 del W3C,** mostra una grammatica in grado di rispondere a comandi del tipo “open file”, “move window”…

```
#ABNF 1.0 UTF-8;
language en;
mode voice;
root $basicCmd;
meta "author" is "Stephanie Williams";
 /**
Basic command.
* @example please move the window
* @example open a file
*/
 public $basicCmd =
                $<http://grammar.example.com/politeness.gram#startPolite>
                $command
                $<http://grammar.example.com/politeness.gram#endPolite>;
$command = $action $object;
$action = /10/ open {TAG-CONTENT-1} | /2/ close {TAG-CONTENT-2}
                | /1/ delete {TAG-CONTENT-3} | /1/ move {TAG-CONTENT-4};
 $object = [the | a] (window | file | menu);
```

Grammatica e calcolo delle probabilità



Riconoscimento e poi?

Una volta terminato il processo di elaborazione di un evento, il SRV può ulteriormente lavorare sui dati ricevuti dall'utente.

Recentemente, la W3C, organo di creazione degli standard per la rete ed il mondo dei computer, ha definito le *Semantic Tags*: queste sono etichette che specificano il valore semantico dei dati presenti nella grammatica.

Ha inoltre creato il *Natural Language Semantics Markup Language* che permette di rappresentare l'output della macchina come sintesi vocale.

Un esempio di Natural language mark up language (NLML)

Il seguente è un esempio di NLML per un SRV dedicato alla ricezione di ordinazioni in una pizzeria inglese…

```
$order = I would like a $drink {$.drink = new Object(); $.drink.liquid = $drink.type;
       $.drink.drinksize = $drink.drinksize}

       and $pizza {$.pizza=$pizza};
          // two properties on $order, both are structs
          // drink was passed property by property to change a property name
          // pizza is passed as whole struct

$kindofdrink = coke | pepsi | "coca cola":"coke";

$foodsize = [ {"medium"} | small | medium | large | regular {"medium"}];
          // medium is default if nothing said $tops = {$=new          Array;}

$top {$.push($top)} (and $top {$.push($top)})<1-> ;
          // construct Array of toppings, return Array $top = anchovies | pepperoni |
          mushroom:"mushrooms" | mushrooms;

$drink = $foodsize $kindofdrink {$.drinksize=$foodsize; $.type=$kindofdrink };
          // two named properties (drinksize and type) on left hand side attribute

$pizza = $number $foodsize {$.pizzasize=$foodsize; $.number=$number} pizzas with
       $tops {$.topping=$tops};
          // three properties on $pizza's attribute

$number = (a | one):"1" | two:"2"| three:"3";
```

Il prodotto dell'NLML

Consideriamo il seguente enunciato:

"I would like a coca cola and three large pizzas with pepperoni and mushrooms."

Su di esso, la grammatica creerebbe una struttura come la seguente:

```
{
drink: {
        liquid:"coke"
        drinksize:"medium"}
pizza: {
        number: "3"
        pizzasize: "large"
        topping: [ "pepperoni", "mushrooms" ]
        }
}
```

E' interessante notare che lo stesso enunciato, espresso in XML, necessiterebbe di circa **70** istruzioni per essere processato!

Interpretazione semantica

L'interpretazione semantica è la nuova frontiera dei SRV.

Attualmente, può avvenire solo in contesti ristretti e può risolvere solo eventi singoli e semplici.

Si riferisce solo a domini specifici, con un numero limitato di eventi possibili.

Ogni lingua ha etichette proprie.

HAL 9000 è ancora molto, molto lontano…

Applicazioni

I SRV hanno trovato moltissime applicazioni:

Supportano la creazione di documenti ed la loro catalogazione.

Permettono il controllo dei computer e delle macchine ad essi collegate, sostituendo i meccanismi manuali di input, sia a livello professionale che ludico..
Sostituiscono l'uomo in compiti ripetitivi come il dare informazioni o il servizio di centralino.

Possono aiutare chi soffre di disabilità motorie o sensoriali.

E' del luglio di quest'anno la notizia che 4 ospedali tedeschi del distretto della Saar hanno installato un SRV che collega tutti i reparti ed aiuta il personale sanitario nella compilazione di documenti e nel lavoro quotidiano, portando ad un significativo incremento nell'efficienza ed al risparmio di oltre il 50% del tempo dedicato a tali compiti.

Gli applicativi

I programmi per il RV più diffusi sono:

ScanSoft Speechwork

ScanSoft Dragon Dictate e Naturally Speaking

IBM ViaVoice

Lernout&Hauspie Voice Xpress

Microsoft Speech Engine

Babel Technologies Babear

Vocalis Speechware

Philips Speech Magic

Il mercato

Tutti i programmi menzionati nella diapositiva precedente hanno una versione dedicate alle aziende ed una dedicata all'uso domestico, con costi e prestazioni diverse.

Il mercato dei SRV è in continua espansione: da meno di un miliardo di dollari previsto per quest'anno, si passerà, entro il 2008, a circa 5 miliardi (fonte Kelsey Group).

Bibliografia e riferimenti

http://www.bridgeport.edu/sed/projects/cs597/Fall_2002/tphilip/

http://www.mor.itesm.mx/~omayora/Tutorial/tutorial.html

http://florin.stanford.edu/~t361/Fall2000/TWeston/home.html

http://research.microsoft.com/srg/

http://www.w3.org/TR/speech-grammar/

http://www.w3.org/TR/semantic-interpretation/

http://www.scansoft.com/speechworks

R. Pieraccini – **"The Art and Science of Spoken Language Systems from Research to Industry" –** Speechwork International

P. Nenad - **Natural Language Processing and Speech Enabled Applications** – University of Sheffield

K. Kemble – **An introduction to speech recognition** – IBM Corporation

CALL Centre – **Speech Recognition Systems –** University of Edinburgh