



UNIVERSITA' DEGLI STUDI DI VENEZIA

Facoltà di Scienze Matematiche, Fisiche e Naturali

Corso di Laurea in Scienze dell'Informazione

TESI DI LAUREA

Un'applicazione per la predizione testuale,  
progettata per persone con disabilità  
neuromotoria

Laureando: Mancin Roberto

Relatore: Ch.mo Prof. Flavio Sartoretto

Correlatore: Ch.mo Prof. Edoardo Arslan

ANNO ACCADEMICO 1996-97

Pare che nel gioco della vita non sia così importante avere delle belle carte quanto il saper giocare bene quelle che si hanno. Dedico questa tesi alle persone che mi hanno insegnato a giocare: mio fratello, mio papà, mia mamma, Edda, Marino, Ada, Laura e nuovamente mio fratello.

## **Ringraziamenti**

Desidero ringraziare di cuore il professor Flavio Sartoretto, che mi ha accolto e seguito con entusiasmo in questo lavoro di tesi, ed il professor Edoardo Arslan per l'attenzione e la disponibilità dimostratemi. Un grazie vivissimo anche al professor Rodolfo Delmonte, per i preziosi consigli e la gentilezza.

Un grazie a Marco Ponza e alla dott.ssa Augusta Rivarola per l'ispirazione, al dott. Silvano Pontello per il finanziamento che mi ha permesso di sviluppare un primo ausilio per la comunicazione, all'associazione "Paolo Bonsembiante" e all'ing. Angelo Boschetti per aver creduto e sostenuto l'utilizzo degli ausili informatici per promuovere l'indipendenza e la partecipazione sociale delle persone con disABILITÀ e ad Adriana Belotti, la mia cavia preferita.



# Indice

INDICE.....	i
ELENCO DELLE TABELLE.....	vi
ELENCO DELLE FIGURE.....	vi
SOMMARIO.....	vii
<b>1 DISABILITÀ E INTEGRAZIONE.....</b>	<b>1</b>
<b>1.1 Menomazione, disABILITÀ ed handicap.....</b>	<b>2</b>
<b>1.2 Ausili adattivi.....</b>	<b>2</b>
<b>1.3 Tipologie di disABILITÀ.....</b>	<b>3</b>
<i>1.3.1 DisABILITÀ visive.....</i>	<i>4</i>
<i>1.3.2 DisABILITÀ motorie (ortopediche).....</i>	<i>4</i>
<i>1.3.3 DisABILITÀ uditive.....</i>	<i>5</i>
<i>1.3.4 DisABILITÀ intellettive.....</i>	<i>5</i>
<i>1.3.5 DisABILITÀ nella comunicazione verbale.....</i>	<i>5</i>
<b>1.4 Patologie neuromotorie che provocano disABILITÀ nella comunicazione verbale... 6</b>	<b>6</b>
<i>1.4.1 Paralisi cerebrali infantili.....</i>	<i>6</i>
<i>1.4.2 Distrofia muscolare.....</i>	<i>7</i>
<i>1.4.3 Sclerosi laterale amiotrofica (ALS).....</i>	<i>7</i>
<i>1.4.4 Sclerosi multipla (MS).....</i>	<i>7</i>
<i>1.4.5 Trauma cranico (TBI).....</i>	<i>8</i>
<i>1.4.6 Lesione midollare.....</i>	<i>8</i>
<i>1.4.7 Ictus cerebrale.....</i>	<i>8</i>
<b>1.5 Modalità e ausili comunicativi.....</b>	<b>8</b>
<b>1.6 Scopo della tesi.....</b>	<b>10</b>
<b>2 ALTERNATIVE AND AUGMENTATIVE COMPUTER MEDIATED COMMUNICATION (AACMC).....</b>	<b>11</b>
<b>2.1 Augmentative and Alternative Communication (AAC).....</b>	<b>11</b>
<i>2.1.1 Low Tech Augmentative Communication Systems.....</i>	<i>12</i>

## INDICE

2.1.2 <i>Medium Tech Augmentative Communication Systems.</i>	13
2.1.3 <i>High Tech Augmentative Communication Systems.</i>	14
<b>2.2 Computer Mediated Communication (CMC).</b>	<b>15</b>
2.2.1 <i>Codifica testuale o multimediale.</i>	15
2.2.2 <i>Informare e comunicare.</i>	16
2.2.3 <i>Comunicazione sincrona o asincrona (orale o scritta).</i>	16
<b>2.3 CMC “Classica”: metodi, problemi e soluzioni.</b>	<b>17</b>
2.3.1 <i>Metodi per la produzione di informazioni testuali.</i>	17
2.3.2 <i>Problemi posti dal modo testuale ASCII.</i>	18
2.3.3 <i>Soluzioni.</i>	19
<b>3 VELOCIZZARE L'INTERAZIONE UOMO-MACCHINA.</b>	<b>21</b>
<b>3.1 Strategie per velocizzare la CMC.</b>	<b>21</b>
<b>3.2 Efficacia dell’ausilio.</b>	<b>23</b>
<b>3.4 Metodi predittivi.</b>	<b>24</b>
3.4.1 <i>Predizione alfabetica.</i>	25
3.4.2 <i>Predizione statistica.</i>	25
3.4.3 <i>Predizione sintattica.</i>	26
3.4.4 <i>Predizione semantica.</i>	26
3.4.5 <i>Espansione di acronimi-abbreviazioni.</i>	26
<b>3.5 Oggetti su cui operano i metodi predittivi.</b>	<b>27</b>
3.5.1 <i>Carattere.</i>	27
3.5.2 <i>Parola (Word Completion).</i>	28
3.5.3 <i>Frase (Word e Multi Word Prediction).</i>	28
<b>4 PREDIZIONE SINTATTICA.</b>	<b>29</b>
<b>4.1 Predizione sintattica forte.</b>	<b>29</b>
4.1.1 <i>Sintassi.</i>	30
4.1.2 <i>Semantica.</i>	30
4.1.3 <i>Context Free Grammar.</i>	31
4.1.4 <i>Context Free Grammar aumentate.</i>	34
<b>4.2 Predizione sintattica debole.</b>	<b>35</b>
4.2.1 <i>Matrice dei pesi.</i>	36

## INDICE

4.2.2	Concordanza .....	37
<b>5</b>	<b>L'ALGORITMO DI PREDIZIONE.....</b>	<b>39</b>
<b>5.1</b>	<b>Strutture di dati.....</b>	<b>40</b>
5.1.1	Lexicon: .....	40
5.1.2	Matrice dei pesi.....	41
5.1.3	Registri.....	41
5.1.4	Array di record o stringhe.....	42
5.1.5	Finestra di editing.....	43
<b>5.2</b>	<b>Analisi dell'algoritmo di predizione.....</b>	<b>43</b>
5.2.1	Stato iniziale.....	43
5.2.2	Recupero di parole ortograficamente valide e copia in un array di record.....	43
5.2.3	Calcolo della probabilità e riordinamento.....	44
5.2.4	Visualizzazione delle predizioni.....	44
5.2.5	Digitazione.....	44
<b>5.3</b>	<b>Funzioni e Procedure varie.....</b>	<b>46</b>
<b>6</b>	<b>MIND READER: TEXT EDITOR CON WORD PREDICTION.....</b>	<b>49</b>
<b>6.1</b>	<b>Obiettivi.....</b>	<b>49</b>
<b>6.2</b>	<b>Strumenti di sviluppo: scelte preliminari.....</b>	<b>49</b>
6.2.1	Ambiente di sviluppo.....	50
6.2.2	Hardware.....	51
<b>6.3</b>	<b>L'implementazione.....</b>	<b>51</b>
6.3.1	Lexicon.....	52
6.3.2	Determinazione automatica della matrice dei pesi.....	53
6.3.3	Interfaccia.....	55
6.3.4	Funzione di Predizione.....	61
<b>7</b>	<b>VALUTAZIONE DELL'EFFICACIA DI MIND READER.....</b>	<b>65</b>
<b>7.1</b>	<b>Criteri di valutazione dell'Usability.....</b>	<b>65</b>
7.1.1	Semplicità dell'uso e dell'apprendimento.....	65
7.1.2	Sicurezza.....	66

## INDICE

7.1.3 Efficacia. ....	66
7.1.4 Efficienza. ....	66
7.1.5 Motivazione. ....	66
<b>7.2 Metodi e strumenti per la valutazione.....</b>	<b>67</b>
7.2.1 Misura dell'efficacia: <i>Keystoke_Saving</i> . ....	67
7.2.2 Misura dell'efficienza: <i>type_rate e time_saving</i> . ....	68
<b>7.3 Ottimizzazione del programma. ....</b>	<b>69</b>
7.3.1 Determinazione della matrice dei pesi. ....	70
7.3.2 Ottimizzazione della concordanza. ....	71
<b>7.4 Risultati dei test.....</b>	<b>72</b>
7.4.1 Efficacia. ....	72
7.4.2 Efficienza. ....	78
<b>7.5 Discussione dei risultati. ....</b>	<b>81</b>
<b>8 SUNTO E CONCLUSIONI. ....</b>	<b>85</b>
<b>8.1 DisABILITÀ e CMC.....</b>	<b>85</b>
<b>8.2 Predizione del testo. ....</b>	<b>85</b>
<b>8.3 Il codice Mind Reader. ....</b>	<b>86</b>
<b>8.4 Conclusioni. ....</b>	<b>87</b>
<b>A - CLASSIFICAZIONE DELLE DISABILITÀ NELLA COMUNICAZIONE VERBALE. ....</b>	<b>89</b>
<b>B - DERIVAZIONE DELLA MATRICE DEI PESI USATA DA MIND READER.....</b>	<b>91</b>
<b>C - SOFTWARE DI WORD PREDICTION IN ITALIANO PER WINDOWS.....</b>	<b>93</b>
<b>D - GLOSSARIO.....</b>	<b>99</b>
<b>BIBLIOGRAFIA. ....</b>	<b>104</b>
<b>INDICE ANALITICO.....</b>	<b>109</b>

## Elenco delle Tabelle.

1.1 Cittadini disABILI nell'Unione Europea [3] .....	4
1.2 DisABILITÀ principale in ragazzi americani di 3-21 anni .....	6
3.1 Probabilità di trovare la parola desiderata fra le 5 suggerite dal sistema in funzione del numero di caratteri digitati.....	21
3.2 Tipologie del vocabolario, in funzione del metodo predittivo .....	27
4.1 Esempio di matrice dei pesi per la lingua italiana.....	36
5.1 Probabilità di trovare la parola desiderata nella lista di predizione, utilizzando diversi metodi predittivi, in funzione del numero di caratteri digitati.....	39
5.2 Esempio di lexicon.....	40
7.1 <i>Keystroke_saving</i> ottenuto applicando il solo algoritmo sintattico di Mind Reader al testo di figura 7.1, variando il numero max di suggerimenti e la matrice dei pesi ..	69
7.2 <i>Keystroke_saving</i> al variare del coefficiente moltiplicativo K.....	70
7.3 <i>Keystroke_saving</i> ottenuto al variare di <b>MaxSugg</b> e del tipo di predizione.....	71
7.4 <i>Keystroke_saving</i> ottenuto ripetendo la digitazione del testo di prova con il lexicon <i>Small_lex</i> non aumentato. ....	72
7.5 Copertura del vocabolario ( <i>hit_rate</i> ) e <i>keystroke_saving</i> al variare del metodo predittivo e del vocabolario usato. <b>MaxSugg</b> = 4.....	73
7.6 <i>Keystroke_saving</i> ottenuto usando la predizione statistica con il lexicon <i>big_lex</i> , ripetendo la digitazione del testo e variando <b>MaxSugg</b> .....	74
7.7 <i>Keystroke_saving</i> ottenuto al variare del numero massimo di suggerimenti e del tipo di predizione utilizzando la funzione di non ripetizione. ....	75
7.8 Tabella riassuntiva dei test di valutazione dell'efficacia di Mind Reader compiuti da una studente con disABILITÀ motoria. ....	77
7.9 Valutazione dell'efficienza di Mind Reader.....	79
7.10 Prestazioni di 5 programmi per la Word Prediction in inglese.....	79
7.11 Efficacia della predizione sintattica nei programmi Windmill e Mind Reader. ....	80
B.1 Esempio di matrice dei pesi per la lingua italiana.....	90
C.1 Caratteristiche dei programmi di Word Prediction in italiano.....	92

## Elenco delle figure.

1.1 Barriere comunicative .....	9
2.1 Tabella alfabetica .....	13
2.2 Tabella alfabetica trasparente .....	13
2.3 Comunicatore alfabetico .....	14
2.4. Comunicatore a 100 caselle .....	14
3.1 Esempio di predizione a carattere .....	26
4.1 Regole di produzione .....	32
4.2 Lexicon .....	32
4.3 Albero di derivazione che genera “il cane tocca il gatto” .....	33
4.4 Regole di produzione di una CFG aumentata .....	34
4.5 Lexicon aumentato .....	34
4.6 Lexicon per una grammatica “debole” tipizzata in maniera incompleta .....	37
5.1 Fasi del processo di predizione .....	45
6.1 Architettura di Mind Reader .....	51
6.2. Struttura delle liste di parole e del lexicon <i>big_lex</i> .....	53
6.3 Diagramma duale (keyboard/mouse) dell’interfaccia .....	55
6.4 Layout semplice di Mind Reader .....	56
6.5 Test layout di Mind Reader .....	58
7.1 Brano di prova usato per i test .....	68
7.2 Testo usato per verificare l’efficacia di Mind Reader .....	76
C.1 Pagina Web di Aurora 2.0 in cui si parla della versione italiana del programma ..	93
C.2 Interfaccia di Dedaduls 2.0 .....	94
C.3 Interfaccia di MAIA .....	94
C.4 Interfaccia completa di Mind Reader 1.0 .....	95
C.5 Interfaccia semplice di Mind Reader 1.0 .....	95

## **Sommario.**

Le patologie neuromotorie creano un grave deficit di comunicazione nelle persone che ne sono affette. Infatti sono colpiti sia il canale privilegiato attraverso il quale avviene lo scambio di informazioni tra gli individui, la voce, che le sue naturali alternative quali la scrittura e la gestualità. La persona affetta da questo genere di patologia viene così a trovarsi in una situazione di isolamento che limita seriamente le sue possibilità di sviluppo umano, sociale e lavorativo.

L'utilizzo di ausili informatici adattivi può rimediare almeno parzialmente a questa situazione. Le applicazioni in grado di velocizzare e di semplificare la digitazione di testi costituiscono un'importante categoria di ausili informatici per la comunicazione: particolarmente utili sono i programmi di "word prediction", che facilitano la battitura dei testi suggerendo all'utente le parole che sta digitando. Programmi di questo tipo sono però poco diffusi in Italia.

Scopo di questa tesi è lo sviluppo di un'applicazione per la "word prediction" in lingua italiana che utilizzi diversi algoritmi per formulare i suggerimenti, e la valutazione dell'efficacia degli algoritmi stessi.

Questa applicazione potrà successivamente interagire con opportuni dispositivi di input (emulatori di mouse, ecc.) o di output (sintesi vocale) per costituire un ausilio comunicativo adattabile alle esigenze personalizzate del singolo.

Sul sito web della Commissione Disabilità e Handicap dell'Università di Padova (<http://www.unipd.it/cdh/>) è possibile trovare la versione ipertestuale (HTML) ed accessibile di questo lavoro di tesi.



## 1 DisABILITÀ e Integrazione.

Il numero di persone con disABILITÀ<sup>1</sup> fisiche è in continuo aumento nelle nostre società. Le cause di questo fenomeno sono molteplici, come l'aumento degli incidenti stradali e sportivi e il diffondersi di nuove patologie. Particolare incidenza hanno la migliorata situazione sanitaria ed i progressi della scienza medica che permettono l'allungarsi della vita media e la sopravvivenza di persone con gravi lesioni o patologie. Si è ridotta la mortalità fra i bambini con gravi menomazioni congenite, ai quali viene garantita una maggior speranza e qualità di vita; le sindromi degenerative vengono sempre più efficacemente rallentate e contrastate; il progresso delle tecniche rianimatorie e riabilitative consente di salvare la vita a persone che hanno avuto lesioni cerebrali o midollari di diversa origine. Se si considerano anche i bambini più piccoli e gli anziani, la percentuale di disABILI arriva al 20% della popolazione.

Contemporaneamente, si evolve velocemente la concezione che la società ha del disABILE: coloro che un tempo venivano considerati come un peso o fenomeni da nascondere, ora vengono accettati come persone, che come tali hanno il diritto di partecipare pienamente alla vita della società e ne costituiscono una risorsa. Mai come oggi le persone con disABILITÀ hanno avanzato con tale determinazione precise richieste per avere le stesse possibilità ed opportunità degli altri cittadini.

Questo cambiamento non è dettato solo da un'esigenza di giustizia, ma anche da una necessità economica: trasformare i disABILI, per quanto è possibile, da soggetti passivi ed assistiti in individui autonomi e produttivi significa liberare un'enorme quantità di risorse umane che potranno contribuire efficacemente allo sviluppo della società.

Questa rivoluzione culturale sta accadendo nello stesso momento in cui tutte le società occidentali stanno entrando nell'Età dell'Informazione; un periodo in cui la conoscenza e l'utilizzo delle tecnologie informatiche e

---

<sup>1</sup> Le parole *disABILITÀ* e *disABILE* verranno scritte sempre in questo modo per ricordare continuamente che la persona che ha una disABILITÀ non è inABILE ma solo diversamente ABILE.

delle informazioni digitali saranno fondamentali per integrarsi attivamente nella società. La nuova barriera che le persone con una disABILITÀ si troveranno ad affrontare sarà proprio il problema dell'accesso a questi strumenti, che vengono progettati per persone in grado di sfruttare al meglio tutte le loro capacità motorie e sensoriali. Più ancora che il non poter camminare o vedere, il non poter utilizzare le tecnologie informatiche sarà la vera fonte di handicap nella società del nostro futuro.

### 1.1 Menomazione, disABILITÀ ed handicap.

Quando ci si muove nel mondo della disABILITÀ si deve far uso di termini che hanno significati ed accezioni diverse. Chiarire il significato delle parole è quindi importante per poter capire esattamente la natura dei problemi di cui ci stiamo occupando e le soluzioni che si possono ricercare. L'Organizzazione Mondiale della Sanità nella "*Chiarificazione delle menomazioni, delle disABILITÀ e degli svantaggi esistenziali*", pubblicata nel 1980 dà le seguenti definizioni:

- **menomazione (impairment):** qualsiasi perdita o anormalità a carico di una struttura o di una funzione psicologica, fisiologica o anatomica.
- **disABILITÀ (disABILITY):** qualsiasi limitazione, conseguente a menomazione, della capacità di compiere un'attività nel modo o nell'ampiezza considerati normali per un essere umano.
- **svantaggio (handicap):** condizione conseguente ad una menomazione o ad una disABILITÀ, che in un certo soggetto limita o impedisce l'adempimento del suo ruolo normale in relazione all'età, al sesso e a fattori socio culturali.

La menomazione di per sé non costituisce un handicap, lo diventa solo quando è causa di svantaggio per la persona che ne è affetta. Chiaramente se si riesce a far sì che una particolare menomazione non diventi causa di svantaggio, si è eliminato conseguentemente anche l'handicap. La via da seguire per ottenere questo risultato è quella degli ausili adattivi.

### 1.2 Ausili adattivi.

La radice latina "auxilium" identifica un "aiuto" in una accezione piuttosto ampia: l'ausilio, in generale, è quella apparecchiatura, attrezzatura o accorgimento che consente alla persona disABILE di

attivare o potenziare una propria autonomia, nel rispetto delle possibilità ed esigenze proprie e dell'ambiente circostante.

Relativamente agli ausili tecnici, per ausilio si intende qualsiasi prodotto, strumento, attrezzatura o sistema tecnologico di produzione specializzata o di comune commercializzazione, utilizzato dalla persona disABILE per prevenire, compensare, alleviare o eliminare una menomazione, disABILITÀ o handicap [1].

Chiariamo con un esempio. I miopi sono persone affette da una menomazione dell'apparato visivo, quindi sono disABILI rispetto all'azione del vedere e handicappati in tutta una serie di attività quotidiane quali il guidare o l'assistere a spettacoli cinematografici. Grazie all'utilizzo dell'ausilio "occhiali" o "lenti a contatto", i miopi possono superare la loro disABILITÀ e quindi evitare l'handicap. Questo esempio riguarda uno strumento estremamente efficace, ma nella maggior parte dei casi gli ausili disponibili sono in grado di compensare solo in parte la disABILITÀ e quindi possono contribuire ad alleviare, ma non ad eliminare, la situazione di handicap.

L'efficacia concreta dell'ausilio dipende molto dalla sua flessibilità, cioè dalla possibilità di modificarlo e configurarlo in modo da rispondere nel modo migliore alle esigenze dell'utente. Un ausilio flessibile viene detto adattivo. Gli ausili informatici, per le loro caratteristiche intrinseche, possono essere progettati in modo da essere adattivi.

### **1.3 Tipologie di disABILITÀ.**

Le patologie che possono essere causa di disABILITÀ sono numerosissime, ma dal punto di vista di chi si occupa dell'applicazione e dello sviluppo di ausili informatici, è molto meno importante sapere qual è la sorgente della menomazione che conoscerne le conseguenze. Una persona può aver perso l'uso della mano destra in seguito a un incidente o a un ictus ma in ogni caso ha bisogno di una periferica di input che le permetta di aggirare questa menomazione [2]. Naturalmente informazioni di tipo medico (patologia e prognosi) e storico (menomazione congenita o acquisita) sono molto importanti per pianificare la riabilitazione ed il reinserimento sociale e lavorativo, ma esulano dagli scopi di questa tesi.

Nell'ambito di questo lavoro ricorriamo ad una classificazione che divide le disABILITÀ in categorie correlate con il tipo di difficoltà di

accesso ai sistemi informatici. In tabella 1.1 sono riportate le diverse categorie e la loro incidenza nei paesi della Comunità Europea.

**Tabella 1.1:** Cittadini disABILI nell'Unione Europea [3].

Tipo di disABILITA'	% della Popolazione	% sul totale dei cittadini della CEE disABILI
Visive	2,0	18
Motorie (Arti Inferiori/superiori)	5,8/1,9	51/17
Uditive	2,7	24
Intellettive	2,3	20
Comunicazione verbale	1,1	10

Ogni diversa categoria, come vedremo più in dettaglio nei prossimi paragrafi, presenta problemi differenti che possono essere affrontati con l'aiuto di ausili informatici.

### 1.3.1 DisABILITÀ visive.

Questa categoria include le persone ipovedenti, daltoniche o completamente cieche ed in generale tutte quelle che hanno dei problemi nel vedere la tastiera e lo schermo e a leggere del materiale stampato. Purtroppo, con l'introduzione delle GUI (graphical user interface) le disABILITÀ visive stanno diventando sempre più delle inABILITÀ.

**Ausili e metodi.** Per l'ipovisione: ingranditori di schermo, occhiali; per il daltonismo: colori configurabili; per la cecità totale: lettore di schermo (software che reindirizza l'output testuale ad una sintesi vocale).

### 1.3.2 DisABILITÀ motorie (ortopediche).

L'accesso al calcolatore è problematico per chi non ha l'uso ottimale delle gambe, solo in presenza di barriere architettoniche. Diverso è il problema per chi ha difficoltà nell'uso degli arti superiori, dal momento che l'utilizzo delle tradizionali periferiche per l'input (tastiera, mouse) può essere impossibile.

**Ausili e metodi:** tastiere speciali, riconoscimento vocale, emulatori di mouse, "caschetto".

### 1.3.3 DisABILITÀ uditive.

Attualmente l'interazione con il calcolatore avviene praticamente solo in modo visivo (per l'output) e fisico (per l'input). La veloce diffusione di sistemi multimediali, quindi anche sonori, o l'utilizzo di sistemi per il riconoscimento vocale per velocizzare l'input possono però creare dei disagi nell'accesso all'informazione per le persone che soffrono di sordità totali o parziali.

**Ausili e metodi:** regolazione del volume, opportuno filtraggio del segnale sonoro, ridondanza visiva delle informazioni sonore.

### 1.3.4 DisABILITÀ intellettive.

Raramente una persona con problemi di memoria o di apprendimento ha difficoltà ad accedere al software ed all'hardware standard, soprattutto se basato su interfacce grafiche.

Al contrario l'impiego del calcolatore nella riabilitazione cognitiva o per ovviare a problemi di interpretazione delle immagini o dei suoni è di enorme utilità e versatilità.

### 1.3.5 DisABILITÀ nella comunicazione verbale.

E' costituita dalle patologie che provocano difficoltà nel produrre un parlato intelligibile. E' sicuramente questa la categoria di in cui le normali abilità comunicative di una persona (verbali e gestuali) sono più seriamente compromesse.

Quasi sempre queste disABILITÀ sono causate dall'incapacità del sistema nervoso centrale di coordinare i movimenti dei muscoli volontari, e quindi interessano anche il resto del corpo. Questo potrebbe quindi sembrare un sottogruppo del terzo, ma l'impossibilità di usare la voce come strumento di input (gli ausili basati sulla voce come metodo di input sono i più efficienti) ne fa una categoria con problemi del tutto peculiari che meritano un ulteriore approfondimento. Una classificazione più rigorosa delle disABILITÀ comunicative è riportata nell'appendice A.

Secondo la tabella 1.1 si tratta di una categoria non molto significativa dal punto di vista numerico. Se limitiamo però l'analisi alle persone in età scolare (3-21 anni), vediamo dalla tabella 1.2 che le disABILITÀ comunicative sono un problema importante nelle fasce più giovani della popolazione. I dati riportati dalle due tabelle, riferendosi a due campioni diversi, permettono un confronto puramente qualitativo ma sono

ugualmente indicativi dato che le patologie invalidanti e le terapie adottate sono estremamente simili nei due casi.

**Tabella 1.2:** DisABILITÀ principale in ragazzi americani di 3-21 anni (fonte: [http://trace.wisc.edu/docs/30\\_some/30\\_some.htm](http://trace.wisc.edu/docs/30_some/30_some.htm))

	% sul totale degli studenti americani	% sul totale degli studenti disABILI
Intellettive	4,2	58
Comunicazione verbale	1,8	25
Altre	1,3	18

## 1.4 Patologie neuromotorie che provocano disABILITÀ nella comunicazione verbale.

Un grave deficit comunicativo impedisce alla persona di interagire normalmente con l'ambiente, con conseguenze importanti per il suo sviluppo sociale, personale ed intellettuale e naturalmente per il suo ingresso nel mondo della scuola e del lavoro. La causa di tale deficit è solitamente collegata ad un insufficientemente grado di destrezza motoria, e quindi di controllo sui muscoli che presiedono alla produzione del parlato, provocato da diversi tipi di patologie neuromotorie. E' importante sottolineare che la scarsa destrezza motoria non colpisce solo la produzione del parlato, ma anche tutti quei movimenti fini utilizzabili per forme alternative di comunicazione, quali ad esempio la scrittura o il linguaggio dei segni. Analizziamo in dettaglio le diverse patologie neuromotorie alla base di questo particolare tipo di disABILITÀ [4].

### 1.4.1 Paralisi cerebrali infantili.

Si tratta di un disordine neuromotorio che colpisce il sistema nervoso centrale nella fase di sviluppo. I problemi motori sono di tipo diverso, a seconda del sito della lesione: alcuni si manifestano con *povertà di movimento* ed altri con *eccesso di movimento e incoordinazione*. Il disordine neurologico ha conseguenze sia per il linguaggio parlato (disartria, disfunzioni respiratorie e laringee) sia per le abilità manuali necessarie ad utilizzare una macchina da scrivere o la tastiera di un computer.

### 1.4.2 Distrofia muscolare.

Malattia ereditaria caratterizzata dalla degenerazione lenta ma progressiva delle fibre muscolari e quindi dalla *perdita di funzione per carenza di forza*. La differenziazione dei bisogni di comunicazione e di controllo dell'ambiente è strettamente legata alla forma di distrofia (ad esempio ai gruppi muscolari maggiormente interessati) e il problema principale è di adeguare lo strumento tecnico all'evoluzione temporale della disABILITÀ.

### 1.4.3 Sclerosi laterale amiotrofica (ALS).

Si tratta di una malattia progressiva e degenerativa che coinvolge i neuroni del cervello e del midollo spinale (famosa per aver colpito l'astrofisico inglese Stephen Hawking [5]). Il sintomo dominante è una debolezza generalizzata progressiva che lascia però intatte le facoltà cognitive. Il problema ricorrente da affrontare è la *progressiva diminuzione dell'ampiezza dei movimenti e la flessibilità della voce*. Con questo tipo di pazienti è necessario utilizzare molteplici ausili di comunicazione adattandoli al progredire della disABILITÀ anche perché sono pazienti che sono in grado di passare da un tipo di comunicazione ad un altro a seconda del contesto e dell'interazione sociale, compatibilmente con il livello di forza disponibile.

### 1.4.4 Sclerosi multipla (MS).

E' una malattia demielinizzante, di solito progressiva, a carico della sostanza bianca del sistema nervoso. I sintomi della malattia variano moltissimo a seconda della localizzazione della lesione. Si osservano *paralisi di tipo spastico agli arti*, deficit di tipo sensoriale, ***incoordinazione*** (atassia), tremori (soprattutto agli arti superiori), talvolta sono presenti *disturbi del linguaggio e della parola*. Presentandosi una così ampia varietà di sintomi, l'indicazione di ausili di comunicazione deve essere individualizzata e adattata ai problemi specifici dei singoli pazienti. Gli ausili sono mirati soprattutto alla facilitazione della comunicazione scritta, là dove sono presenti problemi visivi o agli arti superiori, o della parola.

### **1.4.5 Trauma cranico (TBI).**

E' una patologia legata ad incidenti traumatici che vanno ad interessare l'area cerebrale e colpisce una fascia di popolazione molto giovane. Il manifestarsi dei segni patologici dipende sia dall'entità e vastità del danno cerebrale, sia dal percorso post-traumatico: sono piuttosto comuni i *problemi di tipo cognitivo, comportamentale o di controllo motorio*. E' in quest'ultimo caso che diventa estremamente importante l'utilizzo di un sistema per la comunicazione in grado di adattarsi e di soddisfare tutte le necessità e potenzialità comunicative del paziente, sia nella fase del risveglio dal coma, che in quella della riabilitazione ospedaliera, e soprattutto nel lungo e difficile periodo del reinserimento sociale.

### **1.4.6 Lesione midollare.**

Anche questa disABILITÀ è legata ad eventi traumatici ed è la conseguenza di una lesione del midollo spinale. E' una patologia neuromuscolare, ma l'abilità necessaria per mantenere intelligibile il parlato non viene minimamente compromessa.

### **1.4.7 Ictus cerebrale.**

Colpisce prevalentemente la popolazione anziana. Sono lesioni di parte dell'encefalo, provocate dall'interruzione dell'irrogazione sanguigna o da un'emorragia. Ne risulta un deficit di tipo sensoriale e motorio dell'emisoma corporeo opposto all'emisfero cerebrale colpito. *Il segno patologico più evidente è di tipo motorio*, ma talvolta si osserva afasia se la lesione è avvenuta nell'emisfero dominante, e deficit visuo-percettivi se è avvenuta nell'altro lato. Fino ad ora si è ricorso ad ausili di comunicazione soprattutto come aiuto temporaneo per risolvere problemi di afasia come sistemi di chiamata/allarme per persone con grossi problemi di spostamento.

## **1.5 Modalità e ausili comunicativi.**

La caratteristica fondamentale di tutte le patologie appena esaminate è una diminuita capacità comunicativa.

Uno scambio comunicativo fra due individui è un trasferimento bidirezionale di informazioni fra le loro menti. Affinché questo avvenga è necessario che l'intenzionalità comunicativa dell'uno si trasformi in

## CAPITOLO 1 - DisABILITÀ e Integrazione

un'attività meccanica (parlare, scrivere, mimare) che codifica il messaggio in un formato accessibile ai sensi dell'interlocutore. La comunicazione non è possibile quando i due interlocutori non usano lo stesso protocollo (ad esempio due lingue diverse) oppure quando nel canale attraverso cui avviene la comunicazione sono presenti altri segnali (rumori) che disturbano la comunicazione.

Da un punto di vista modellistico, una disABILITÀ di tipo neuromotorio può essere vista come un fenomeno che riduce la capacità di un canale di comunicazione: il compito dell'ausilio è quello di aumentare l'efficacia dell'utilizzo di questo canale accettando in input un segnale prodotto dalla persona disABILE e producendo come output un segnale più pulito, intelligibile e manipolabile.

Restringendo ulteriormente il campo ai sistemi informatici, diciamo che un ausilio per la comunicazione deve essere in grado di interfacciarsi opportunamente con la persona, affinché questa possa dirigere la generazione di un testo ASCII che può essere facilmente utilizzato da una qualsiasi applicazione standard.

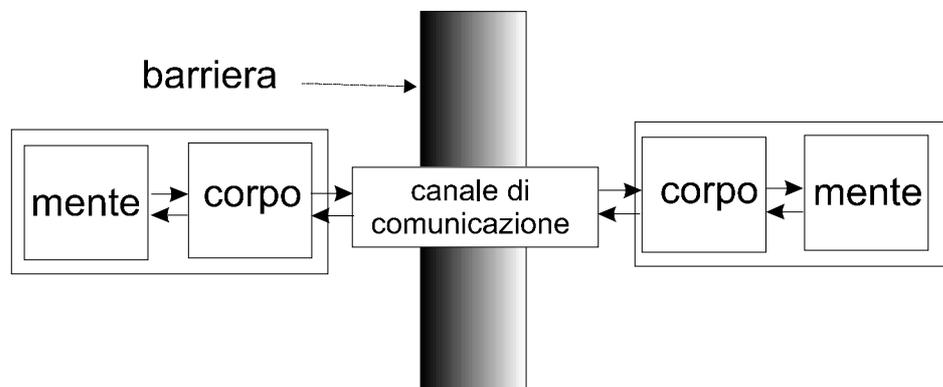


Figura 1.1: Barriere comunicative

## **1.6 Scopo della tesi.**

Le patologie neuromotorie, come abbiamo visto, creano un grave deficit di comunicazione nelle persone che ne sono affette. Infatti sono colpiti sia il canale privilegiato attraverso il quale avviene lo scambio di informazioni tra gli individui, la voce, che le sue naturali alternative quali la scrittura e la gestualità. La persona affetta da questo genere di patologia viene così a trovarsi in una situazione di isolamento che limita seriamente le sue possibilità di sviluppo umano, sociale e lavorativo.

L'utilizzo di ausili informatici adattivi può rimediare almeno parzialmente a questa situazione. Scopo di questo lavoro è lo sviluppo di un'applicazione in grado di velocizzare e di semplificare la digitazione di testi.

Questa applicazione potrà successivamente interagire con opportuni dispositivi di input (emulatori di mouse, ecc.) o di output (sintesi vocale) per costituire un ausilio comunicativo adattabile alle esigenze personalizzate del singolo.

L'applicazione sviluppata è stata sottoposta ad una serie di test teorici e applicativi, volti a verificarne la reale efficacia nell'aumentare la velocità e la qualità della comunicazione di persone affette da patologie neuromuscolari.

## **2 Alternative and Augmentative Computer Mediated Communication (AACMC).**

E' nota l'importanza dell'integrazione sociale dei cittadini con disABILITÀ. Questo obiettivo può essere raggiunto facendo ricorso ad ausili informatici capaci di operare una riduzione dell'handicap.

Particolarmente grave è il problema di integrare persone con deficit comunicativi di origine neuromotoria: questa disABILITÀ provoca infatti l'esclusione di queste persone dall'ambiente sociale, scolastico e lavorativo. In questo ambito, l'utilizzo di ausili informatici permette di superare ostacoli altrimenti insormontabili.

### **2.1 Augmentative and Alternative Communication (AAC).**

Interagire con le cose e con le persone, giocare, studiare, comunicare sono necessità vitali per lo sviluppo umano ed intellettuale.

Le persone con gravi menomazioni fisiche vengono gravemente handicappate non solo dalle loro difficoltà oggettive ma anche da una serie di luoghi comuni che le descrivono come soggetti inattivi: può comunicare solo chi può parlare, può viaggiare solo chi può camminare, può scrivere solo chi può usare la penna, può vedere solo chi ha occhi efficienti e può partecipare attivamente alla vita solo chi non ha gravi menomazioni fisiche. Forse un tempo tutto questo era vero, ma oggi la tecnologia ci mette a disposizione degli strumenti che permettono di superare o attenuare gli effetti handicappanti provocati da ogni menomazione.

Nel campo dei deficit comunicativi la strada da seguire è quella dei sistemi di comunicazione alternativi ed aumentativi: metodi per amplificare le capacità comunicative di una persona facendo ricorso a canali alternativi di trasmissione dei messaggi.

Il primo e forse più diffuso sistema alternativo di comunicazione è il linguaggio dei segni, utilizzato da molte persone sorde ma anche da chi,

pur non avendo problemi fonatori, vuole aggiungere maggiore enfasi ed efficacia al parlato.

L'idea di utilizzare il corpo per comunicare dei concetti si perde nella notte dei tempi e molto probabilmente è precedente allo sviluppo del linguaggio nell'uomo. Bisogna però attendere gli anni del secondo dopo guerra perché l'Augmentative and Alternative Communication [6] diventi un vero e proprio settore di ricerca e le prime applicazioni tecnologiche vengano sviluppate: durante la guerra, infatti, un gran numero di persone aveva acquisito una disABILITÀ e nello stesso periodo si iniziava a favorire l'inserimento dei bambini con menomazioni congenite (vedi capitolo 1) [7].

Negli anni seguenti il linguaggio dei segni ed i sistemi di comunicazione che si basano su simboli (LIS e Bliss) divennero una vera e propria alternativa al linguaggio parlato per la comunicazione interpersonale. Questi metodi non risolvono tutti i problemi delle persone con deficit comunicativi, in particolare in ambito didattico e lavorativo [8]

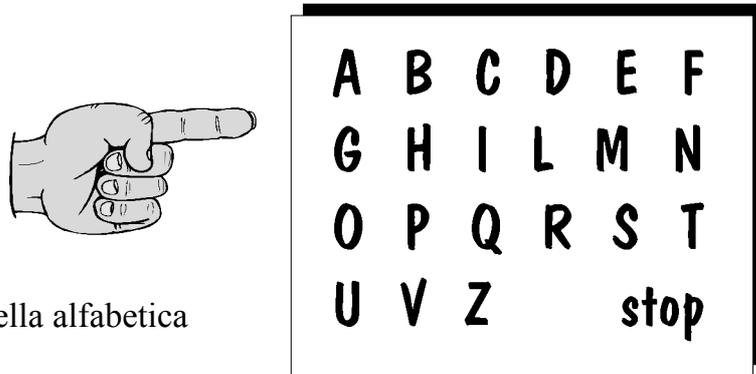
Numerosi sono attualmente gli ausili e i metodi, disponibili sul mercato od oggetto di ricerca, sviluppati ed utilizzati per facilitare alle persone disABILI la comunicazione, anche perché sono diversissime le abilità residue e le necessità di chi ne ha bisogno.

Nei paragrafi che seguono esamineremo alcuni di questi metodi in ordine di crescente complessità [9].

### **2.1.1 Low Tech Augmentative Communication Systems.**

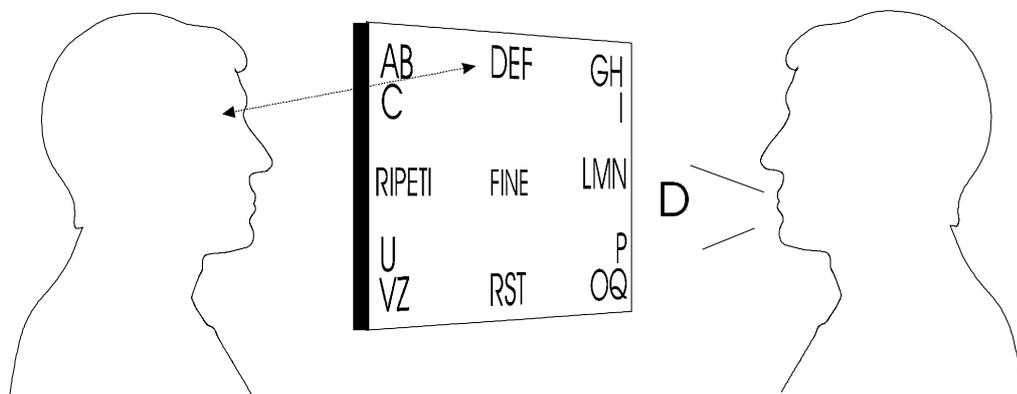
Spesso non è necessario e nemmeno opportuno usare ausili sofisticati e costosi per sopperire ad un deficit comunicativo. Se la disABILITÀ è temporanea, o se le esigenze comunicative del soggetto sono limitate, l'ausilio ideale è costituito da una semplice tabella alfabetica di cartone o di plexiglass come quella illustrata nella figura. 2.1: l'utente indica in successione le lettere che compongono le parole che vuole formare servendosi delle mani o della direzione dello sguardo. Spesso l'interlocutore è in grado di indovinare la parola basandosi sulle prime lettere, e questo contribuisce a velocizzare la comunicazione. Alcuni metodi particolari possono essere utilizzati per semplificare l'uso della tabella nel caso che l'utente debba servirsi dello sguardo come indicatore. Ad esempio in figura 2.2 le lettere sono raggruppate in gruppi di 3 e spetta

all'interlocutore capire quale di lettere l'altro vuole selezionare, basandosi sulle lettere precedenti e servendosi di un feedback verbale.



**Figura 2.1:** tabella alfabetica

E' bene ricordare che la scelta di usare ausili così semplici è una conseguenza della situazione contingente (urgenza, semplicità, ...) e non è per nulla in relazione con le abilità intellettive di chi ne fa uso. Semmai è vero il contrario, cioè che ausili più costosi e sofisticati devono essere prescritti quando è necessario aumentare la facilità di utilizzo del sistema, a causa dei deficit cognitivi dell'utente.



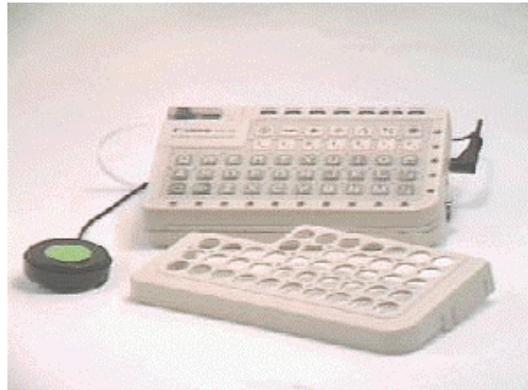
**Figura 2.2:** tabella alfabetica trasparente.

### 2.1.2 Medium Tech Augmentative Communication Systems.

In alcuni casi, sistemi per l'AAC artigianali come i precedenti sono più che sufficienti, ad esempio quando lo scopo è solo quello di comunicare le necessità più urgenti a medici, infermieri e familiari. Se però le necessità sono maggiori, divengono necessari sistemi tecnologicamente più sofisticati, che possano permettere la trasmissione di messaggi più

complessi e articolati. Tali ausili devono essere portatili, dotati di diverse modalità di input (tastiera, scansione, ecc.) e di output (display luminosi, stampa su carta, voce sintetica o digitale). I principali rappresentanti di questa categoria sono:

- **comunicatori alfabetici:**  
macchine da scrivere elettriche o elettroniche miniaturizzate e portatili: l'utente compone su una tastiera i messaggi che poi vengono stampati su strisce di carta o riprodotti con voce sintetica. La velocità della comunicazione è comunque estremamente bassa.



**Figura 2.3:** Comunicatore Alfabetico

- **comunicatori simbolici:**  
apparecchi elettronici dotati di 16 o 100 caselle, selezionabili mediante una luce pilotata dall'utente, contenenti le icone di un linguaggio simbolico non verbale. In linguaggi di questo tipo concetti e idee sono espressi mediante simboli pittografici: la comunicazione è veloce, ma i concetti esprimibili sono semplici e l'interlocutore deve conoscere il linguaggio (vedi anche paragrafo 3.1).



**Figura 2.4:** Comunicatore a 100 caselle

### 2.1.3 High Tech Augmentative Communication Systems.

Negli ultimi anni, grazie al crollo dei prezzi e la miniaturizzazione dei calcolatori è diventato più economico trasformare un normale computer portatile in ausilio per la comunicazione attraverso opportune modifiche all'hardware ed al software.

Rispetto agli ausili a media tecnologia descritti nel paragrafo 2.1.2, il computer utilizzato in qualità di ausilio per la comunicazione, presenta una maggior flessibilità e versatilità, ma anche una minore affidabilità (è più facile il verificarsi di guasti o malfunzionamenti).

E' inoltre importante sottolineare che servendosi di sistemi informatici l'utente può sfruttare anche tutte le caratteristiche di un normale computer: ad esempio può utilizzare le possibilità comunicative offerte dall'uso di Internet e della posta elettronica.

### **2.2 Computer Mediated Communication (CMC).**

Il più potente ausilio per la comunicazione è un computer opportunamente equipaggiato; rendere accessibili alle persone con disABILITÀ i normali sistemi informatici e telematici significa fornire non solo un formidabile strumento di comunicazione e di relazione, ma anche di lavoro e di istruzione.

Quando parliamo di "accessibilità" dei mezzi informatici e telematici dobbiamo sempre tenere presente con quale modalità (multimediale o testuale), a che scopo (comunicare o informare), e con quali tempi (dialogo in tempo reale o scambio di messaggi) questi vengono usati.

#### **2.2.1 Codifica testuale o multimediale.**

Normalmente l'uomo si serve di tutti i sensi per ricevere e trasmettere messaggi. Le interfacce dei computer, fino a poco tempo fa, erano solo di tipo testuale: si trattava di una soluzione povera ma facilmente gestibile. Le informazioni provenienti da un unico canale possono essere riversate con poche difficoltà su un altro canale: un output testuale, naturalmente accessibile attraverso la vista, può essere letto (convertito in suono) o trasmesso da una barra braille (convertito in informazioni tattili) senza alcuna perdita di informazioni e senza la necessità di particolari accorgimenti da parte dell'autore.

Negli ultimi anni lo sviluppo delle tecnologie informatiche ha reso disponibili e ampiamente utilizzate interfacce di tipo multimediale, che ormai sono diventate la forma standard di output. Questo sviluppo rappresenta un notevole avvicinamento alla naturale comunicazione umana: la macchina fornisce all'utente informazioni servendosi contemporaneamente di più canali, rendendole così più comprensibili e

appetibili. Si creano però problemi di accessibilità: chi non può utilizzare efficientemente tutti i canali sensoriali viene privato di una parte del messaggio, ed è molto più difficile riversare queste informazioni perse su canali alternativi che sono già occupati dalle altre parti del messaggio. La soluzione di questo problema richiede un impraticabile ritorno alle interfacce testuali o l'adozione di un formato multimediale ridondante: tutte le informazioni disponibili su tutti i canali di trasmissione [10].

### **2.2.2 Informare e comunicare.**

Informare significa trasmettere messaggi ad un grande numero di persone in maniera non interattiva (monodirezionale). Il grosso problema è consentire l'accesso all'informazione: rendere fruibili testi, immagini e suoni anche da chi ha disABILITÀ sensoriali (visive e uditive), ossia rendere universalmente utilizzabili gli output visivi e sonori. La comunicazione (bidirezionale) richiede invece lo scambio di informazioni tra due o più persone. Il formato con cui vengono trasmesse le informazioni è più semplice (parole, testi) ma oltre all'accessibilità dell'output è importante anche l'accessibilità e l'efficienza dell'input. In particolare è necessario velocizzare il modo in cui una persona con disABILITÀ motoria può utilizzare il calcolatore, ad esempio per scrivere testi.

I problemi dell'accessibilità dell'output e dell'input sono solitamente trascurati o sottovalutati: facilitare e velocizzare la comunicazione tramite il computer (Computer Mediated Communication) è difficile.

E' un problema tecnologico perché richiede l'utilizzo di dispositivi opportunamente progettati a seconda delle esigenze dell'utente e algoritmi di intelligenza artificiale non ancora implementati negli attuali sistemi informatici. Ed è un problema economico perché richiede la modifica (accessible design) o una nuova progettazione (universal design) dei sistemi utilizzati.

### **2.2.3 Comunicazione sincrona o asincrona (orale o scritta).**

Bisogna distinguere due modi fondamentali di comunicare, che richiedono modalità e strumenti decisamente diversi:

- ❑ comunicazione in tempo reale (orale).
- ❑ comunicazione differita (scritta).

Il requisito più importante della comunicazione orale, il dialogo tra due o più persone, è la velocità di trasmissione e la fruibilità immediata del messaggio da parte del ricevente. A questa esigenza si possono sacrificare anche requisiti di "correttezza": ad esempio sono ammessi gli errori di ortografia o si può non terminare le parole. Come abbiamo visto nel paragrafo 2.1, la comunicazione può servirsi di diversi canali di uscita come la voce, i gesti, le immagini, la scrittura. Non è importante la forma del prodotto finale, quanto il fatto che il messaggio arrivi effettivamente a destinazione nel più breve tempo possibile. Perciò il messaggio può ridursi al puntamento da una lista di lettere, parole o frasi, può essere visualizzato su un tabellone o un display luminoso, o può utilizzare una voce preregistrata.

Abbiamo visto inoltre che la comunicazione in tempo reale implica sempre l'uso di un linguaggio, ma non necessariamente di quello verbale comunemente utilizzato. Si possono quindi avere anche delle forme di comunicazione interpersonali non-verbali come il linguaggio dei segni o il Bliss.

La comunicazione scritta richiede un'impostazione totalmente diversa rispetto a quella orale. Molto spesso la produzione del testo prevede tempi più lunghi (si pensi ad un compito in classe, una lettera, un testo professionale ecc.), ma il prodotto finale deve essere più preciso, tipicamente accompagnata da un'uscita su carta. Il computer è praticamente l'unico strumento utilizzabile per permettere ai disABILI un tale tipo di comunicazione.

### **2.3 CMC “Classica”: metodi, problemi e soluzioni.**

Analizziamo ora in dettaglio la CMC “classica”, intesa come scambio **testuale** e asincrono fra due interlocutori.

#### **2.3.1 Metodi per la produzione di informazioni testuali.**

Fino a poco tempo fa le interfacce dei calcolatori erano di tipo testuale e l'accesso avveniva grazie al codice ASCII (*American Standard Code for Information Interchange*), che è attualmente lo standard più diffuso, più accessibile e più pratico per gestire le informazioni elettroniche. Successivamente, abbiamo avuto una rapida diffusione delle GUI (*Graphic User Interface*) e dei sistemi multimediali. Prima che ciò avvenisse la

stragrande maggioranza dei sistemi operativi, dei programmi applicativi, delle risorse e dei servizi telematici (e-mail, Gopher, Telnet, Usenet) utilizzavano lo standard ASCII per rappresentare le informazioni.

Una caratteristica importantissima della codifica di un testo effettuata seguendo lo standard ASCII è la completa indipendenza dal tipo di hardware e di software utilizzato per l'accesso. Il testo può essere facilmente visualizzato su un monitor, su una barra braille o letto da una scheda per la sintesi vocale. E' facilissimo trasformare l'informazione digitale in una rappresentazione visiva, tattile o sonora.

### **2.3.2 Problemi posti dal modo testuale ASCII.**

Il problema principale che si presenta quando si vuole rendere possibile ad un disABILE la comunicazione tramite computer in modo testuale ASCII è quello dell'input: inserire dati nel sistema scegliendo fra un numero limitato di alternative non è difficile, ma comunicare significa partecipare attivamente ad una discussione, articolando ed esponendo in tempi brevi i propri pensieri. Attualmente le interfacce uomo macchina più efficienti sono: tastiera, mouse, e voce (riconoscimento vocale). Quando questi non sono utilizzabili a causa di una disABILITÀ, è possibile adottare strategie ed ausili alternativi per interagire con la macchina, ma si tratta di soluzioni lente e complesse, come la scansione. Anche quando si superano le difficoltà fisiche con opportuni ausili, la lentezza diventa un grave handicap, in particolare se la conversazione avviene in tempo reale.

Un altro problema dell'ASCII è che in realtà è veramente universale solo a 7 bit, ma fra i 128 simboli diversi ( $2^7=128$ ) che codifica non ci sono le lettere accentate. Gli standard a 8 bit ( $2^8=256$ ) le prevedono, ma non esiste attualmente uno standard universale a 8 bit usato in tutte le nazioni e da tutti i software. Ad esempio MS-DOS e Windows fanno al riguardo scelte diverse: questo spiega perché le lettere accentate di un file di testo creato in un certo ambiente a volte risultano stranamente trasformate se il file è letto nell'altro.

Un ultimo problema è l'impossibilità di distinguere velocemente le informazioni importanti da quelle secondarie, per un utente che debba esaminare una grossa quantità di testi. In ASCII infatti non è possibile sottolineare una frase o variare la dimensione dei caratteri per evidenziare un passaggio.

### 2.3.3 Soluzioni.

Fortunatamente i problemi dell'accessibilità ai testi sono noti da molto tempo e gran parte delle difficoltà possono essere risolte con poca fatica utilizzando piccoli accorgimenti nella produzione degli stessi: ad esempio l'utilizzo corretto della punteggiatura, che rende più chiara la lettura automatica, l'evitare tabelle e grafici non descritte nel testo e note a piè pagina contenenti informazioni necessarie per la comprensione. E' bene inoltre limitare l'uso di corsivo, grassetto e sottolineato ed utilizzare piuttosto le virgolette per dare maggiore enfasi ad una frase.

Di più difficile soluzione è il problema della velocizzazione dell'input. Teoricamente il controllo volontario di un solo muscolo è sufficiente per accedere, tramite opportuni ausili hardware (switch) e software (scansione), a qualsiasi risorsa informatica. Purtroppo un accesso di questo tipo è estremamente lento e frustrante, e spesso non è sufficiente per soddisfare le necessità di una persona con disABILITÀ.

Quando la comunicazione è asincrona, la velocità non è un elemento handicappante. Il ricorso a metodi di comunicazione di questo tipo, come la posta elettronica, si potrebbe quindi rivelare di grande aiuto per le persone con disABILITÀ.

Quando invece la comunicazione è sincrona, la soluzione potrebbe venire dall'impiego di algoritmi di predizione, in grado di velocizzare la generazione del testo ASCII, riducendo il numero di tasti premuti per comporre una frase.



### 3 Velocizzare l'interazione uomo-macchina.

Il problema maggiore che incontra nell'accesso alla CMC chi ha un deficit comunicativo causato da una patologia neuromotoria è dovuto alla difficoltà nel coordinare i muscoli: codificare l'intenzionalità comunicativa generando una sequenza di azioni intelligibili dal calcolatore risulta un compito estremamente impegnativo.

Molte risorse umane ed economiche vengono spese per sviluppare dispositivi utilizzabili da chi non può usare le normali periferiche per l'input quali tastiera, mouse e microfono. Minore invece è lo sforzo che si fa per cercare di rendere efficienti questi dispositivi.

Con l'impiego di opportuni ausili tecnologici e informatici (vedi paragrafo 1.3.2) è possibile aggirare e superare gran parte delle barriere fisiche che ostruiscono il canale attraverso il quale il messaggio, generato nella mente della persona con disABILITÀ di origine neuromotoria, può giungere al computer. Purtroppo molte di queste soluzioni informatiche sono così lente da renderne l'uso estremamente noioso: un ausilio per la comunicazione che non tiene conto di questo fattore rischia di essere inutilizzabile. L'accesso al calcolatore, infatti, non è il fine della riabilitazione, ma un mezzo per la riduzione dell'handicap. Quanto più è efficiente la strategia adattiva usata, tanto più è facilitata l'integrazione della persona con disABILITÀ.

#### 3.1 Strategie per velocizzare la CMC.

In questa discussione esamineremo in dettaglio i metodi e gli algoritmi che possono aumentare la quantità di informazioni che l'utente produce nell'unità di tempo.

Quattro sono le principali soluzioni che vengono comunemente adottate per velocizzare la comunicazione per mezzo dei computer:

- **Semantic encoding:** le parole ed i concetti di uso più frequente vengono associati a simboli pittografici. La codifica del messaggio è molto rapida ed efficiente, ma solo se l'utente e l'interlocutore conoscono bene il sistema comunicativo e il messaggio è semplice.

### CAPITOLO 3 - Velocizzare l'interazione Uomo-Macchina

- **Abbreviation expansion:** il sistema espande automaticamente poche lettere digitate, seguendo le istruzioni ricevute in precedenza. Esempi: *DS* → *distinti saluti*, *TVB* → *ti voglio bene*. Come nel primo caso, questo sistema è molto efficiente, anche perché l'utente non deve spostare lo sguardo dalla tastiera al monitor; raramente però una persona è in grado di ricordare più di 100 abbreviazioni e quindi la comunicazione consentita è necessariamente povera.
- **Phrase completion:** il sistema suggerisce la conclusione della frase che si sta digitando. E' utile se lo scritto è ricco di ripetizioni, come nel caso di un codice sorgente, o per le parti iniziali e terminali di una comunicazione.
- **Word completion e prediction:** man mano che l'utente digita una parola, il sistema genera una lista di proposte fra cui suppone che ci sia quella che l'utente sta digitando. Spesso con "word prediction" si intende sia il completamento della parola che si sta digitando (completion o completamento), sia la predizione della parola successiva a quella appena digitata (prediction vera e propria).

**Tabella 3.1:** Probabilità di trovare la parola desiderata fra le 5 suggerite dal sistema in funzione del numero di caratteri digitati.

Numero di caratteri digitati	0	1	2	3
Completamento	0,2%	14,5%	47,7%	92,9%
Predizione+Completamento	29,3%	68,5%	93,2%	99,3%
<b>Efficacia relativa<sup>a</sup></b>	122,2	4,7	2,0	1,1

a) Rapporto tra le probabilità di successo di due metodi

Per chiarire meglio l'importanza di questa sottile distinzione esaminiamo alcuni dati relativi all'efficacia della predizione del testo, estrapolati da un lavoro di M. E. Wood [11]. Dai dati della tabella 3.1 si può vedere come l'efficacia della predizione, rispetto al semplice completamento, diminuisce man mano che l'utente digita la radice

della parola. La prima è 4,7 volte più efficace dopo la digitazione del primo carattere, ma egualmente efficace dopo tre caratteri. La predizione è enormemente efficace quando non è stato digitato nessun carattere della parola successiva: in questo caso la scelta dei suggerimenti è guidata dalla sola conoscenza della parola appena digitata, ma tale informazione non viene usata nel completamento.

Le ultime tre soluzioni (abbreviation expansion, phrase completion, word prediction), che possono essere implementate contemporaneamente nello stesso programma allo scopo di migliorarne l'efficienza, rientrano nell'insieme dei metodi di predizione. Ad ogni digitazione di un tasto, il sistema propone un elenco di parole che presumibilmente completano la parola (o la frase) pensata dall'utente. Ogni alternativa di questa lista viene associata ad un tasto funzionale. Premendo quel tasto, l'intera alternativa viene inserita nel testo, a partire dal punto in cui si trova il cursore. Le proposte nell'elenco verranno chiamate anche *suggerimenti*.

Un parametro importante per valutare l'utilità di questi metodi è, come vedremo nel capitolo 7, il *keystroke\_saving*: la percentuale di caratteri risparmiati (non digitati) sul totale dei caratteri visualizzati.

$$\text{keystroke\_saving} = \left( 1 - \frac{\text{tasti\_premuti}}{\text{caratteri\_visualizzati}} \right) * 100$$

Nei capitoli seguenti verranno discusse, dal punto di vista teorico ed algoritmico, la **word completion** e la **word prediction**, ne verrà inoltre verificata l'utilità per la scrittura di testi in italiano attraverso lo sviluppo di un'applicazione.

### 3.2 Efficacia dell'ausilio.

Per determinare la reale utilità di un ausilio dobbiamo porci tre domande, in relazione ad altrettanti aspetti:

- **aspetto ergonomico**: L'ausilio è adattabile al campo di variazione delle variabili motorie (escursione, velocità, forza...) dell'utente?

## CAPITOLO 3 - Velocizzare l'interazione Uomo-Macchina

- **aspetto economico:** Sono stati minimizzati i costi materiali e cognitivi dovuti all'apprendimento, l'utilizzo e l'eventuale riparazione?
- **aspetto psicofisico:** Abbiamo minimizzato il rapporto fra il numero di movimenti volontari che l'utente può produrre nell'unità di tempo e la quantità di output generato?

E' difficile rispondere a queste tre domande poiché le esigenze ergonomiche, le potenzialità informativo-comunicative e la motivazione dei soggetti disABILI sono molto variabili nel tempo.

La scelta di opportuni metodi (*scansione a n tasti, joystick, mouse, tastiera*) e ausili per l'input (*switch meccanici, termici, ottici, tastiere speciali, emulatori di mouse*), fatta con la supervisione di uno specialista (terapista occupazionale), può portare ad identificare ausili che rispondono al primo criterio. Per soddisfare il secondo criterio è importante cercare una soluzione compatibile con gli ambienti operativi più diffusi.

I software di predizione che implementano alcune delle soluzioni elencate nel paragrafo 3.1 danno vita ad un ausilio che risponde al terzo criterio.

### 3.3 Metodi predittivi.

I metodi di predizione sono raggruppabili in quattro categorie, a seconda del metodo utilizzato per suggerire la parola.

- **Alfabetico:** propone i suggerimenti utilizzando solo l'ordine alfabetico come criterio di selezione.
- **Statistico:** genera i suggerimenti sulla base di informazioni statistiche relative alla frequenza d'uso della parola. Questa categoria può essere ulteriormente suddivisa in due sotto-gruppi: con vocabolario statico o con vocabolario dinamico.
- **Sintattico:** usa informazioni sintattiche nel tentativo di incrementare l'accuratezza della previsione prodotta.
- **Semantico:** si usa il contenuto semantico dell'ultima frase digitata per effettuare la predizione. Gli algoritmi e le risorse richieste non sono oggi praticabili.

E' importante notare che i metodi statistici, sintattici e semantici non sono fra loro alternativi, anzi l'impiego contemporaneo di metodi ridondanti permette risultati migliori. Vediamo in dettaglio le metodologie.

### **3.3.1 Predizione alfabetica.**

Il sistema si limita a suggerire in ordine alfabetico tutte le parole che cominciano con le lettere già digitate. Si tratta di un algoritmo estremamente semplice, ma inefficiente.

### **3.3.2 Predizione statistica.**

La quasi totalità dei sistemi attualmente realizzati impiegano metodi di questo tipo. La scelta delle parole da suggerire è guidata dalla probabilità che hanno i caratteri o le parole di apparire in un testo. Il valore di probabilità può essere legato ad analisi linguistiche fatte a priori, e quindi essere fisso, oppure può dipendere dallo stile dell'utente e modificarsi man mano che il sistema viene usato.

#### **Dizionario statico.**

Si tratta del sistema di predizione statistica più semplice. Ad ogni lemma viene associata una costante che indica la frequenza con cui questa parola viene utilizzata. In genere queste associazioni fra parola e frequenza vengono prodotte analizzando testi provenienti da libri: questo metodo crea problemi quando il linguaggio utilizzato dall'utente ha caratteristiche diverse da quello letterario, ad esempio è di tipo colloquiale o tecnico.

#### **Dizionario dinamico.**

Il dizionario si aggiorna con l'uso del programma: se la parola digitata non è già presente, viene aggiunta nel dizionario, altrimenti viene incrementato il contatore che ne indica la frequenza. In questo modo il sistema impara il linguaggio utilizzato dal proprio utente e può dare i suggerimenti più appropriati.

#### **Dizionario multiplo (bigrammi e trigrammi).**

E' nota al sistema la probabilità di ogni coppia o terna di lettere o parole, cioè la probabilità di una parola, nota la seguente o le due precedenti. Se adattato allo stile dell'utente può essere un metodo efficiente, ma richiede una maggiore quantità di risorse (questo metodo è usato soprattutto per il riconoscimento vocale).

### **3.3.3 Predizione sintattica.**

Distinguiamo due tipi di predizione sintattica.

**Forte** (grammaticale).

L'italiano è una lingua ben strutturata e definita ed impone regole abbastanza rigorose nella costruzione della frase. Anche se è possibile scrivere "il dottore sono buoni", è legittimo aspettarsi che l'utente voglia costruire solo frasi corrette dal punto di vista sintattico. L'applicazione delle regole della sintassi permette al sistema di ridurre il numero delle parole che possono essere usate e quindi di migliorare il suggerimento.

**Debole** (tipizzata).

Ad ogni lemma del dizionario è associata una variabile che ne indica la categoria sintattica (aggettivo, avverbio, verbo). Il sistema di predizione utilizza questa informazione per suggerire la parola successiva. Ad esempio se l'utente ha appena digitato un articolo è estremamente improbabile che anche la parola successiva appartenga a questa stessa categoria sintattica. E' possibile stabilire la probabilità di una categoria sintattica, data la categoria della parola precedente.

### **3.3.4 Predizione semantica.**

La conoscenza dell'argomento di una conversazione rende possibile la selezione di parole appartenenti al gruppo semantico del contesto. Per esempio se l'argomento della discussione è "nautico", allora parole come "nave", "mare", "vela", saranno più probabili di parole come "carbone", "bambola", "salvadanaio". In un contesto didattico, l'argomento di un corso determina automaticamente la scelta di certe parole. L'implementazione di questa tecnica di predizione è proibitiva, in termini di risorse se si vuole realizzare un ausilio per la comunicazione generico. E' certamente sensato realizzare diversi dizionari da utilizzare in determinate situazioni comunicative, in modo che la selezione del contesto semantico (effettuata dall'utente e non dalla macchina) possa condizionare positivamente la predizione.

### **3.3.5 Espansione di acronimi-abbreviazioni.**

Stringhe di testo di uso frequente possono essere prememorizzate dall'utente, associandole ad una breve sequenza di tasti. Questo non è un metodo predittivo vero e proprio, ma implementato all'interno di un

programma di predizione ne migliora in modo sensibile il *keystore\_saving*.

### 3.4 Oggetti su cui operano i metodi predittivi.

#### 3.4.1 Carattere.

La predizione più semplice, nel caso che si utilizzi un input a scansione, è quella che si può fare sul carattere successivo, in base alla frequenza assoluta dei caratteri o in funzione dei caratteri precedenti. E' possibile, ma non molto utile, pre-processare l'elenco di caratteri proposti in modo da eliminare quelli che darebbero stringhe impossibili come gho, que, ecc.



**Figura 3.1:** esempio di predizione a carattere

### 3.4.2 Parola (Word Completion).

Il sistema di predizione cerca di indovinare quale parola si sta scrivendo, non appena si digitano i primi caratteri. Questo processo può essere velocizzato se vengono selezionate quelle grammaticalmente corrette.

(es. i bambini      ~~buone~~  
                          buoni  
                          ~~buone~~  
                          buona)

### 3.4.3 Frase (Word e Multi Word Prediction)

Il sistema non si limita ad indovinare la parola che si sta digitando, ma cerca anche di predire le successive attraverso un'analisi statistica e/o sintattica del testo.

Un notevole miglioramento dell'efficacia di questo tipo di predizione si potrebbe ottenere attraverso un'analisi semantica del testo in modo automatico. Come abbiamo già detto, anche se questo metodo richiederebbe l'impiego di tecniche di programmazione e algoritmi troppo complessi, è possibile comunque delimitare il campo semantico delle parole suggeribili selezionando manualmente un opportuno dizionario.

In ogni caso, questa tesi non svilupperà la Multi Word Prediction.

**Tabella 3.2:** Tipologie del vocabolario, in funzione del metodo predittivo (colonne) e del tipo di oggetto su cui questo opera (righe).

	alfabetico	statistico	sintattico	abbr-exp
carattere	statico	statico		
parola	statico	dinamico/statico	dinamico	dinamico/statico
frase	statico	dinamico/statico	dinamico	dinamico/statico

## 4 Predizione Sintattica.

Mentre si scrive un testo, la scelta delle parole che si vogliono scrivere non è casuale: anche se dipende in primo luogo dalla volontà e dalla capacità di scegliere del soggetto, è condizionata da numerosi altri fattori quali il contesto semantico, le regole ortografiche e della sintassi, lo stile personale dell'utente. Questi fattori possono essere sfruttati dal software di word prediction per ottenere una predizione migliore di quella effettuata con i soli metodi statistici. Attualmente i sistemi per la comunicazione non implementano ottimamente tutte queste funzionalità.

Un programma realistico deve essere in grado di fare delle ipotesi sulla parola che si sta per digitare, tenendo conto di alcune informazioni grammaticali e lessicali ricavabili in modo automatico: tanto più le ipotesi si avvicinano alla volontà comunicativa del soggetto, tanto maggiore è l'efficienza dell'ausilio.

Spesso i sistemi di predizione esistenti suggeriscono parole scorrette dal punto di vista grammaticale. Questo costituisce uno svantaggio: le parole scorrette sottraggono spazio ai suggerimenti validi, e l'utente è costretto perdere del tempo per leggerle e scartarle.

Lo scopo della predizione sintattica è quello di superare questo inconveniente, eliminando le parole errate: ai tradizionali algoritmi di analisi statistica viene fornito come input solamente il sottoinsieme del dizionario costituito dalle parole sintatticamente corrette. In tal modo vengono rimosse le parole presumibilmente inutili, lasciando più spazio a quelle corrette e aumentando l'efficacia della predizione, ossia il risparmio del numero di tasti premuti (*keystroke saving*).

### 4.1 Predizione sintattica forte.

Un sistema in grado di utilizzare conoscenze linguistiche e grammaticali per guidare la predizione può essere progettato utilizzando analizzatori linguistici formali.

Una frase non è una semplice sequenza di parole: esistono delle strutture sintattiche intermedie, come i *periodi*, che a loro volta possono essere suddivisi in sottogruppi, fino ad arrivare alle singole parole. Una

descrizione di questa metodologia per l'analisi della frase è stata presentata da Noam Chomsky nei suoi lavori, *Syntactic Structures* e *Aspects of the Theory of Syntax* [12, 29].

#### 4.1.1 Sintassi

La sintassi è un'insieme di regole per costruire frasi. La mente umana può facilmente riconoscere quando una frase obbedisce a queste regole, anche se queste variano con il contesto. Per esempio si considerino le due frasi:

1. *La carota russa l'uomo nel negozio che stanno vedendo.*
2. *L'uomo che russa vede la carota nel negozio.*

La prima frase è scorretta dal punto di vista sintattico ed è difficile assegnarle un significato. Viceversa la frase 2 è sintatticamente corretta.

Le regole che governano la struttura delle proposizioni all'interno delle frasi e l'ordine delle parole all'interno delle proposizioni sono note come *regole di sottocategorizzazione*. Le frasi che violano queste regole sono dette ad *alto grado di devianza*.

#### 4.1.2 Semantica

La semantica è costituita da *regole di selezione*. Esse sono più difficili da definire delle regole di sottocategorizzazione sintattiche. Questo è dovuto al fatto che spesso è possibile interpretare frasi che disobbediscono alle regole di selezione, inserendole in un contesto differente. Si prenda per esempio la frase seguente

*Davide fu inghiottito dalla notte*

Il soggetto è “la notte” ed il complemento oggetto è “Davide”. E' Davide che viene inghiottito e la notte che sta compiendo l'azione dell'inghiottire. Ma la notte è un concetto astratto e quindi non può compiere alcuna azione. Quindi questa frase viola le regole di selezione, anche se soddisfa quelle di sottocategorizzazione.

Ma se ci si trova in un contesto metaforico, questa frase può significare che Davide è sparito dalla nostra vista, nel buio della notte.

Una grammatica che consente una descrizione analitica completa del linguaggio deve tenere entrambi gli aspetti in considerazione. Chomsky sostiene che nel creare il modello di una lingua è possibile considerare separatamente la sintassi e la semantica. Questa assunzione rende possibile l'utilizzo della teoria dei linguaggi formali per costruire un modello della lingua che tenga in considerazione solo le regole sintattiche. Un modello di questo tipo può essere utilizzato per analizzare o predire le possibili strutture di una frase.

#### 4.1.3 Context Free Grammar.

L'approccio di Chomsky prevede l'utilizzo di una *Grammatica Libera dal Contesto*, una struttura ad albero per rappresentare il modo in cui una frase può essere suddivisa in periodi e come da questi possono nascere ulteriori sottogruppi. Si tratta di un sistema semplice, ma potente, che può essere usato da un calcolatore per automatizzare la manipolazione di un testo scritto. Le Grammatiche Libere dal Contesto costituiscono il formalismo comunemente usato per interpretare (compilare) i linguaggi formali.

#### Definizione formale di CFG.

Una Grammatica Libera dal Contesto (CFG: Context Free Grammar), o più semplicemente una grammatica, è una quartupla:

$$G = (V, T, P, S)$$

dove **V** e **T** sono insiemi, finiti e disgiunti, rispettivamente di *variabili* (gruppi sintattici) e di *terminali* (categorie sintattiche in cui sono suddivise le parole). **S** è una speciale variabile chiamata *simbolo di start*. **P** è un insieme finito di produzioni (regole di derivazione) del tipo  $A \rightarrow \alpha$ , dove **A** è una variabile ed  $\alpha$  è una stringa di simboli di  $(V \cup T)^*$ , cioè di variabili e terminali. Nelle CFG non sono permesse regole del tipo  $\beta \rightarrow \alpha$ , in cui  $\beta$  è una stringa che contiene anche alcuni terminali: in altre parole non sono ammessi simboli terminali nel "lato sinistro" della regola [13].

Vediamo per esempio una CFG in grado di analizzare la frase “il cane tocca il gatto”: Si consideri  $G'=(V,T,P,Frasede)$ , dove  $V=\{Frasede, Parte\_Nominale, Parte\_Verbale\}$ ,  $T=\{articolo, nome, verbo\_transitivo\}$  e  $P$  è l’insieme di regole elencato in figura 4.1:

Frasede  $\rightarrow$  PN + PV  
 PN  $\rightarrow$  articolo + nome  
 PV  $\rightarrow$  verbo transitivo + PN

**Figura 4.1:** Regole di produzione

- il, la, i, le = articolo
- gatto, cane, aquile, petali, carne = nome
- tocca, mangia, volano = verbo\_transitivo

**Figura 4.2:** lexicon

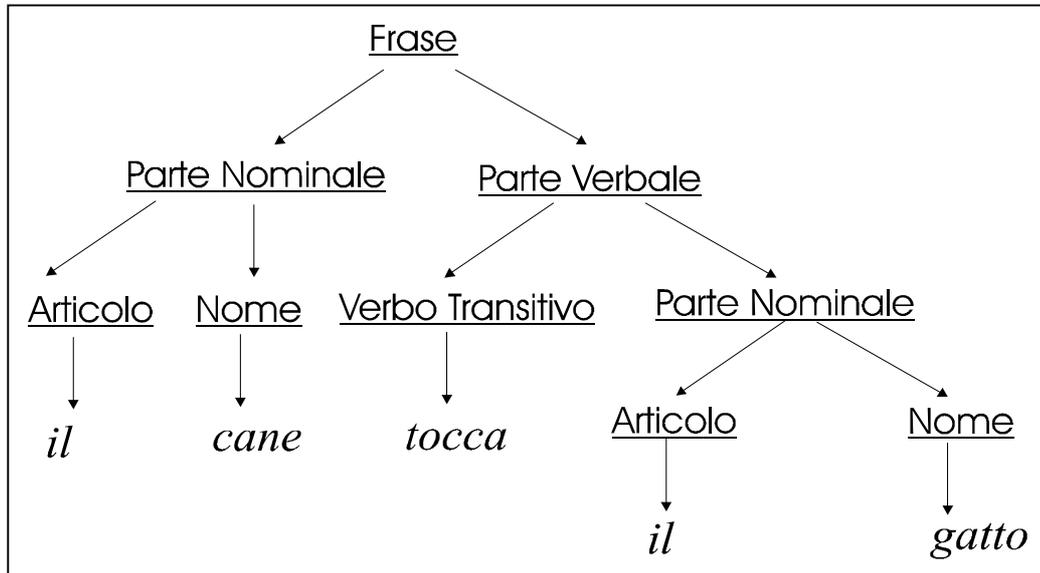
Per generare delle frasi da questa grammatica è necessario un *lexicon*: una struttura di dati che associa ad ogni simbolo terminale in  $T$  una o più parole.

A questo punto è possibile utilizzare la CFG e il lexicon appena definiti in questo modo:

Frasede

- $\rightarrow$  Parte\_Nominale + Parte\_Verbale
- $\rightarrow$  articolo + nome + Parte\_Verbale
- $\rightarrow$  articolo + nome + verbo\_transitivo + Parte\_Nominale
- $\rightarrow$  articolo + nome + verbo\_transitivo + articolo + nome
- $\rightarrow$  il + nome + verbo\_transitivo + articolo + nome
- $\rightarrow$  il + cane + verbo\_transitivo + articolo + nome
- $\rightarrow$  il + cane + tocca + articolo + nome

→ *il + cane + tocca + il + nome*



→ *il + cane + tocca + il + gatto*

**Figura 4.3:** albero di derivazione che genera la frase *il cane tocca il gatto*.

Si indica con  $L(G')$  l'insieme che comprende tutte le frasi che possono essere costruite utilizzando la CFG  $G'$  ed il lexicon dato.

Molte delle sequenze generate non sono corrette dal punto di vista della sintassi italiana (es. la petali volano le fiori) o del significato "naturale" (es. la carne mangia il gatto). Infatti le CFG generano una struttura ad albero che si sviluppa dall'alto verso il basso: in questo modo non esiste nessuna relazione tra la parola scelta per sostituire un determinato simbolo terminale e quella scelta per sostituire il seguente. E' quindi possibile sostituire alla sequenza di terminali articolo+nome sia la sequenza di parole "il cane" che quella "i cane".

Nei linguaggi naturali invece ogni parola può influenzare non solo il tipo delle seguenti, ma anche il genere ed il numero. Per creare una frase sintatticamente corretta bisogna quindi procedere da sinistra a destra: la grammatica deve conoscere e ricordare alcune informazioni legate alle parole precedenti, per coniugare correttamente i verbi o per suggerire le corrette concordanze. Una CFG non è in grado di assolvere a questa

funzione dato che, come abbiamo visto, ogni termine è visto come un'entità indipendente.

Nel paragrafo successivo verrà illustrato un metodo per ovviare a questo problema.

#### 4.1.4 Context Free Grammar aumentate.

La composizione di una frase è un processo che avviene da sinistra a destra: per questo motivo la coniugazione del verbo concorda con il soggetto o l'aggettivo è dello stesso genere del sostantivo. Nel paragrafo precedente si è visto che le CFG non sono in grado di trasmettere queste informazioni tra i gruppi sintattici interessati poiché ogni gruppo è completamente indipendente.

Questo problema può essere superato attraverso l'impiego di regole di derivazione *aumentate*: la scelta di una particolare regola è determinata dal tipo di concordanza attesa. Le regole della CFG G' possono essere riscritte:

Frase	→	PN(singolare)+PV(singolare)
Frase	→	PN(plurale)+PV(plurale)
PN(singolare)	→	articolo (singolare, maschile)+nome (singolare, maschile)
PN(singolare)	→	articolo (singolare, femminile)+nome (singolare, femminile)
PN(plurale)	→	articolo (plurale, maschile) +nome (plurale, maschile)
PN(plurale)	→	articolo (plurale, femminile)+nome (plurale, femminile)
PV(singolare)	→	verbo_transitivo (singolare)+PN (singolare)
PV(plurale)	→	verbo_transitivo (plurale)+PN (plurale)

**Figura 4.4:** regole di produzione di una CFG aumentata.

Anche il lexicon deve essere aumentato in modo da codificare le informazioni necessarie

il = art (sing, masch)	cane = nome (sing, masch)
la = art (sing, femm)	aquile = nome (plur, femm)
i = art (plur, masch)	carne = nome (sing, femm)
le = art (plur, femm)	tocca = v_t (singolare)
petali = nome (plur, masch)	mangiano = v_t (plurale)
gatto = nome (sing, masch)	volano = v_t (plurale)

**Figura 4.5:** lexicon aumentato.

Usando questo metodo, la generazione di frasi generate è garantita: il sistema è in grado di seguire la creazione di un testo corretto usando regole statiche. Non viene eliminata la generazione di frasi semanticamente scorrette. Vi sono però forti controindicazioni a questo approccio. La lingua italiana è estremamente complessa e articolata: può quindi essere descritta solo da un insieme di regole di produzione altrettanto complesso e da un lexicon tipizzato in maniera estremamente dettagliata. L'implementazione dell'algoritmo di predizione basato su una CFG aumentata è quindi molto complessa. La creazione di un lexicon sufficientemente ampio richiede una grossa mole di lavoro, inoltre è difficile e scomodo per l'utente rendere dinamico il vocabolario (ogni nuova parola incontrata deve essere analizzata grammaticalmente dall'utente).

Infine la predizione sintattica fornisce prestazioni negative quando la frase generata dall'utente non rispetta le regole sintattiche implementate. Questa eventualità si verifica se l'utente usa uno stile colloquiale, e se decide di velocizzare ulteriormente la comunicazione, omettendo dal testo alcune parole necessarie alla correttezza sintattica ma non alla comprensione del messaggio (ad es. scrivendo "Vado casa" al posto di "Vado a casa").

**4.2 Predizione sintattica debole.**

Una strada alternativa alla precedente consiste nell'usare un metodo probabilistico di predizione sintattica. Questo metodo, meno complesso, non implementa una vera e propria grammatica generativa come le CFG, ma una struttura molto meno rigida e quindi più facilmente adattabile in modo dinamico. È il metodo usato dai prodotti Profect e Syntax PAL [14], per generare le predizioni.

La grammatica viene rappresentata tramite una tabella di pesi a doppia entrata, consultando la quale è possibile sapere qual è la probabilità dei tipi della parola successiva, noto il tipo della precedente. Dall'analisi del

tipo, del genere e del numero il sistema vaglia inoltre le opportune concordanze.

Le informazioni provenienti dalla matrice dei pesi e dall'analisi delle concordanze vengono utilizzate per modificare i valori di frequenza delle parole provenienti dall'analisi statistica: le parole che hanno una maggiore probabilità di essere sintatticamente corrette verranno suggerite per prime. In questo modo il sistema rimane molto flessibile perché permette, anche se vi assegna una minor probabilità, la scelta di parole scorrette o non concordanti.

Il lexicon richiede una tipizzazione meno dettagliata che nel caso della predizione sintattica forte. Il sistema è inoltre in grado di operare anche con parole non tipizzate o parzialmente tipizzate: in questo caso non opera alcuna correzione sintattica sui valori di frequenza associati alla parola sconosciuta oppure utilizza un'analisi automatica per la determinazione della concordanza.

Questo approccio è più facilmente automatizzabile, e privilegia l'efficacia, rispetto alla precisione ed alla correttezza grammaticale.

#### 4.2.1 Matrice dei pesi.

La scelta del tipo sintattico di una parola è parzialmente determinata dalle scelte precedenti, e questo legame è tanto più vincolante quanto più le parole sono vicine. Ad esempio se la parola appena digitata è un verbo, è molto più probabile che la successiva sia un avverbio piuttosto che un altro verbo.

**Tabella 4.1:** Esempio di matrice dei pesi per la lingua italiana

	v.	avv.	prep.	conj.	art.	n.	pro.	agg.	altro
verbo	1	1	1	1	1	1	1	1	1
avverbio	1	0	1	1	1	1	1	1	1
preposizione	1	1	0	1	1	2	1	1	1
congiunzion.	1	1	1	0	1	1	1	1	1
articolo	1	1	1	1	0	2	1	1	1
nome	1	1	2	1	2	2	1	2	1

pronome	1	1	1	2	1	1	0	1	1
aggettivo	1	1	1	1	1	2	1	0	1
altro	1	1	1	1	1	1	1	1	1

Questo concetto può essere espresso numericamente assegnando una probabilità (peso) elevata alla coppia di categorie verbo-avverbio e bassa a quella verbo-verbo. Assegnando una probabilità ad ogni coppia di categorie, si ottiene una matrice di pesi che costituisce il cuore del sistema di predizione. Sfortunatamente non esistono in letteratura tavole di frequenza ricavate dall'analisi sintattica di un corpus di testi italiani sufficientemente ampio.

#### 4.2.2 Concordanza

Un'altra informazione che può essere dedotta dall'analisi della parola precedentemente digitata riguarda il numero e il genere della parola attesa: dopo alcuni tipi di parole è lecito aspettarsene altre dello stesso genere e numero. Il sistema rispetta questo schema aumentando di un fattore stabilito la probabilità delle parole con la giusta concordanza .

Anche questo sistema non è rigido, e privilegia l'efficacia sulla precisione. Quando non si dispone di un lexicon completamente tipizzato, può essere tollerata un'ulteriore semplificazione che permetta di automatizzare, anche se con qualche imprecisione, la derivazione delle informazioni riguardo al genere ed al numero di un nome o di un aggettivo. Ad esempio si può assumere che i vocaboli maschili e singolari terminino per 'o', i singolari e femminili per 'a', i plurali e maschili per 'i', i plurali e femminili per 'e'.

Un breve lexicon è riportato in figura 4.6:

il=art(o)	cane = nome(o)
la=art(a)	aquile = nome
i=art(i)	albero = nome
le=art(e)	tocca = v_t
petali=nome(i)	mangiano = v_t
gatto=nome(o)	volano = v_t

## CAPITOLO 4 - Predizione Sintattica

**Figura 4.6:** lexicon per una grammatica “debole” tipizzata in maniera incompleta. La lettera fra parentesi indica la codifica del genere e del numero della parola, quando questa informazione non è presente il programma la ricava dalla parola stessa.

## 5 L'algoritmo di predizione.

Nel capitolo 3 abbiamo descritto alcuni sistemi “predittivi” e nel capitolo 4 ci siamo ulteriormente soffermati sui metodi sintattici. Non esiste un metodo che rende gli altri superflui, anzi usare più metodi contemporaneamente rende la predizione più efficace.

Nella tabella seguente, che riporta la percentuale di successo della predizione ottenuta dal programma WindMill [11], si vede che l'efficacia della predizione è tanto maggiore quante più sono le informazioni su cui il sistema si può basare.

**Tabella 5.1** Probabilità di trovare la parola desiderata nella lista di predizione, utilizzando diversi metodi predittivi<sup>a</sup>, in funzione del numero di caratteri digitati. Programma WindMill [11, pag 138].

Numero di caratteri digitati	0	1	2	3
Sintattico+Statistico+Completamento <sup>a</sup>	29%	68%	93%	99%
Statistico+Completamento	19%	56%	85%	98%
Sintattico+Completamento	4%	26%	67%	97%
Alfabetico+Completamento	0%	14%	47%	92%

a) si vedano i paragrafi 3.1 e 3.4 per la descrizione dei metodi

Alcuni accorgimenti, che non fanno parte degli algoritmi predittivi, accrescono ulteriormente il `keystroke_saving`: l'inserimento di uno spazio dopo un carattere di punteggiatura o alla fine di una parola espansa dal sistema, rendere maiuscola la prima lettera dopo un punto, ma soprattutto l'espansione di acronimi. Tali accorgimenti vengono implementati in tutti i sistemi di predizione.

Nel processo di predizione che è stato realizzato dallo scrivente, e che ora verrà illustrato, vengono utilizzati contemporaneamente i metodi di predizione statistica e sintattica debole e alcuni accorgimenti per migliorare il `keystroke_saving`.

## 5.1 Strutture di dati.

Una parte fondamentale del programma è costituito da un insieme di strutture di dati interdipendenti, utilizzate dall'algoritmo di predizione per creare una lista di suggerimenti.

### 5.1.1 Lexicon:

Le applicazioni che implementano una conoscenza non superficiale della lingua con cui devono operare, come le applicazioni di riconoscimento vocale, di sintesi vocale o di word prediction, richiedono un dizionario apposito.

Poiché l'efficienza di un programma di Word Prediction è determinata soprattutto dalla qualità delle informazioni che può usare per determinare le parole da suggerire, è fondamentale dedicare particolare cura alla realizzazione ed alla successiva adattività del dizionario. Ad ogni parola devono essere associati il tipo sintattico e la frequenza relativa. Nella nostra implementazione abbiamo aggiunto due campi opzionali per indicare la concordanza (il genere ed il numero del nome o dell'aggettivo che ci si attende dopo determinate parole, campo `ConcAtt`) e l'espansione automatica (un acronimo viene sostituito automaticamente da una parola estesa o da una frase campo `exp/sost`).

**Tabella 5.2:** esempio di lexicon

parola	tipo	conc_att	freq	exp/sost
il	art.	o	1024	
la	art.	a	1122	
i	art.	i	870	
le	art.	e	800	
petali	nome		60	
gatto	nome		110	
cane	nome		120	
aquile	nome		70	
tocca	verbo		27	
mangiano	verbo		23	
volate	verbo		30	
rm	acronimo		0	Roberto Mancin
ds	acronimo		0	distinti saluti

### 5.1.2 Matrice dei pesi.

La seconda struttura di dati fondamentale nel processo di predizione sintattica è una matrice dei pesi, i cui valori indicano la probabilità di ciascuna coppia ordinata di tipi sintattici (la probabilità che ad una parola di una certa categoria ne segua una di un'altra). La determinazione dei pesi è un processo molto delicato poiché da questi dipende fortemente la validità del metodo sintattico implementato.

In prima approssimazione è possibile utilizzare una matrice empirica analizzando dei semplici modelli di frase. Scegliendo come struttura standard di una frase la seguente:

parte nominale + parte verbale + complementi

ed espandendo ogni parte in questo modo:

parte nominale → articolo + [aggettivo] + sostantivo

parte verbale → verbo + avverbio

complemento → preposizione + parte nominale

si ottiene un modello da cui ricavare la matrice dei pesi. Nella tabella 4.6 sono visualizzati i valori ottenuti assegnando peso 2 ad ogni coppia di tipi sintattici che appare nel modello (articolo-aggettivo, articolo sostantivo, ecc.), assegnando peso 1 a tutte le altre possibili coppie e peso a 0 alle coppie di parole dello stesso tipo (articolo-articolo, verbo-verbo).

Una strategia differente consiste nello scegliere i pesi automaticamente: il programma da noi sviluppato è dotato di una funzione che analizza un testo abbastanza lungo e da questo ricava i valori della matrice.

### 5.1.3 Registri.

Le informazioni necessarie per elaborare la lista dei suggerimenti vengono immagazzinate dal programma in alcune variabili globali (registri) in modo da essere immediatamente disponibili.

Questi registri, che possono avere anche valore nullo e vengono aggiornati in fasi diverse del processo, sono:

**ParCur** (Lettere digitate della parola corrente): costituiscono la parte iniziale delle parole suggerite. Questa è l'informazione più importante per il completamento: dopo alcuni caratteri digitati è in

grado di vincolare da sola il suggerimento. Naturalmente non ha valore nella fase di predizione vera e propria, quando non è stato digitato ancora alcun carattere.

**TipoPrec (Tipo parola precedente):** contiene il tipo della parola appena digitata. Questa informazione serve a calcolare la probabilità che la parola successiva appartenga ad un certo tipo sintattico, usando la matrice dei pesi.

**ConcAtt (Concordanza attesa):** contiene il valore assunto dal campo `ConcAtt` della parola appena digitata. Questo campo può anche essere nullo per due diversi motivi: il tipo di parola non prevede una concordanza (avverbi, congiunzioni, ecc.), oppure questa, anche se possibile, non è stata definita. In quest'ultimo caso è possibile tentare di dedurre la concordanza dall'analisi dell'ultimo carattere della parola appena digitata.

Questa informazione consente il calcolo delle parole con il giusto genere e numero. Se la parola digitata è una congiunzione o un verbo, il registro mantiene il valore che aveva assunto per la parola precedente per permettere frasi del tipo “**la casa rossa e bianca**” o “**il bambino è buono**”.

**MaxSugg (Numero massimo di proposte):** il numero massimo di proposte che il sistema presenta. Una lunga lista di suggerimenti richiede tempo per essere esaminata, ma fra pochi suggerimenti vi è meno probabilità di trovare quello giusto. Ogni utente può trovare il giusto compromesso impostando il valore di questo record.

**Save (Numero minimo di caratteri risparmiabili):** il sistema non suggerisce le parole che permettono di risparmiare meno caratteri di quelli definiti: non ha molto senso sprecare posto nella finestra di predizione e tempo per leggere parole che se anche esatte consentirebbero un basso `keystoke_saving`.

### 5.1.4 Array di record o stringhe.

Il file con il lexicon risiede nella memoria non volatile che è più capiente e meno sensibile ad eventuali mal funzionamenti. La lista delle parole plausibili, invece, viene trascritta nella RAM in un array, a cui è facile accedere in modo casuale ed applicare un algoritmo di ordinamento.

Un array di stringhe contiene la lista delle parole già suggerite durante la digitazione della parola corrente. Utilizzando questa lista il programma può evitare di proporre nuovamente parole che l'utente ha già esaminato e scartato.

### **5.1.5 Finestra di editing.**

La finestra in cui viene visualizzato il testo digitato viene gestita dal programma come un array di stringhe, in cui è possibile inserire od aggiungere sottostringhe.

A differenza di tutte le precedenti strutture di dati, quest'ultima è visibile dall'utente. E' importante curare aspetti quali la grandezza, e il tipo, dei caratteri nella finestra di editing, la possibilità di spostarsi con il cursore all'interno del testo e di gestire automaticamente la spaziatura e la punteggiatura. Come vedremo in dettaglio nel capitolo successivo, la necessità di rendere facile l'uso e la manipolazione del testo visualizzato nella finestra di editing è uno dei fattori che ha determinato la scelta del linguaggio di sviluppo.

## **5.2 Analisi dell'algoritmo di predizione.**

Il processo di creazione di una lista di suggerimenti è schematizzato nella figura 5.1: dopo la digitazione di un carattere, il programma estrae dal lexicon una lista di parole compatibili, la riordina in base a criteri statistici e sintattici e suggerisce le prime parole all'utente. La qualità del risultato è correlata positivamente, come si può vedere nella tabella 5.1, con la quantità di informazioni su cui il sistema può fare affidamento.

### **5.2.1 Stato iniziale.**

Quando si deve ancora scrivere la prima lettera, le informazioni relative al tipo e alla concordanza della parola attesa sono nulle, quindi vengono suggerite le parole più frequenti in assoluto. Queste vengono calcolate una sola volta non appena il programma viene lanciato.

### **5.2.2 Recupero di parole ortograficamente valide e copia in un array di record.**

Quando le prime lettere di una parola sono state digitate, inizia il ciclo di predizione: le lettere contenute in `ParCur` vengono utilizzate per

limitare il numero delle parole del lexicon su cui il programma deve lavorare.

Viene selezionato dal lexicon il sottinsieme (sublexicon) delle parole che iniziano con quei caratteri: su queste saranno applicati metodi statistici e sintattici per creare la lista dei suggerimenti da mostrare all'utente .

### 5.2.3 Calcolo della probabilità e riordinamento.

Il sublexicon così creato viene elaborato assegnando al campo `Prob` di ciascun record il valore che si ottiene dalla seguente espressione:

dove `freq` è il valore di frequenza associato alla ogni parola, `peso[a,b]` il valore della matrice dei pesi,  $K \geq 1$  un coefficiente variabile che aumenta la probabilità delle parole concordanti: l'influenza del valore di  $K$  sull'efficacia delle predizioni è stata testata sul campo (paragrafo 7.3.2).

$$\text{prob} = \text{freq} * \text{peso}[\text{tipo\_prec}, \text{tipo}] * K$$

Effettuata l'elaborazione, il sublexicon viene riordinato utilizzando il campo `prob` come chiave.

### 5.2.4 Visualizzazione delle predizioni.

Ottenuto il sublexicon riordinato, il sistema elimina le parole già suggerite in questo ciclo di predizione (se è attiva la funzione *non ripetere*, paragrafo 5.2.5) e quelle che non soddisfano la condizione:

$$\text{length}(\text{Parola}) - \text{length}(\text{Par\_Cur}) \geq \text{Save}$$

quindi mostra all'utente la lista delle più probabili tra le parole rimaste (la lunghezza di questa lista è determinata dal valore contenuto nel registro `MaxSugg`).

### 5.2.5 Digitazione.

A questo punto l'utente decide se digitare un carattere o selezionare una delle predizioni. Nel primo caso possono essere introdotti due tipi di caratteri: se viene digitata una lettera alfabetica [a..z], il sistema aggiorna

la lista delle parole già suggerite ed il registro ParCur e riparte dal punto 5.2.1; altrimenti considera terminata la parola, aggiorna il lexicon e comincia un altro ciclo. Nel secondo caso viene chiamata la procedura che inserisce la parola proposta nel punto in cui c'è il cursore e quindi si dà inizio ad un nuovo ciclo di predizione.

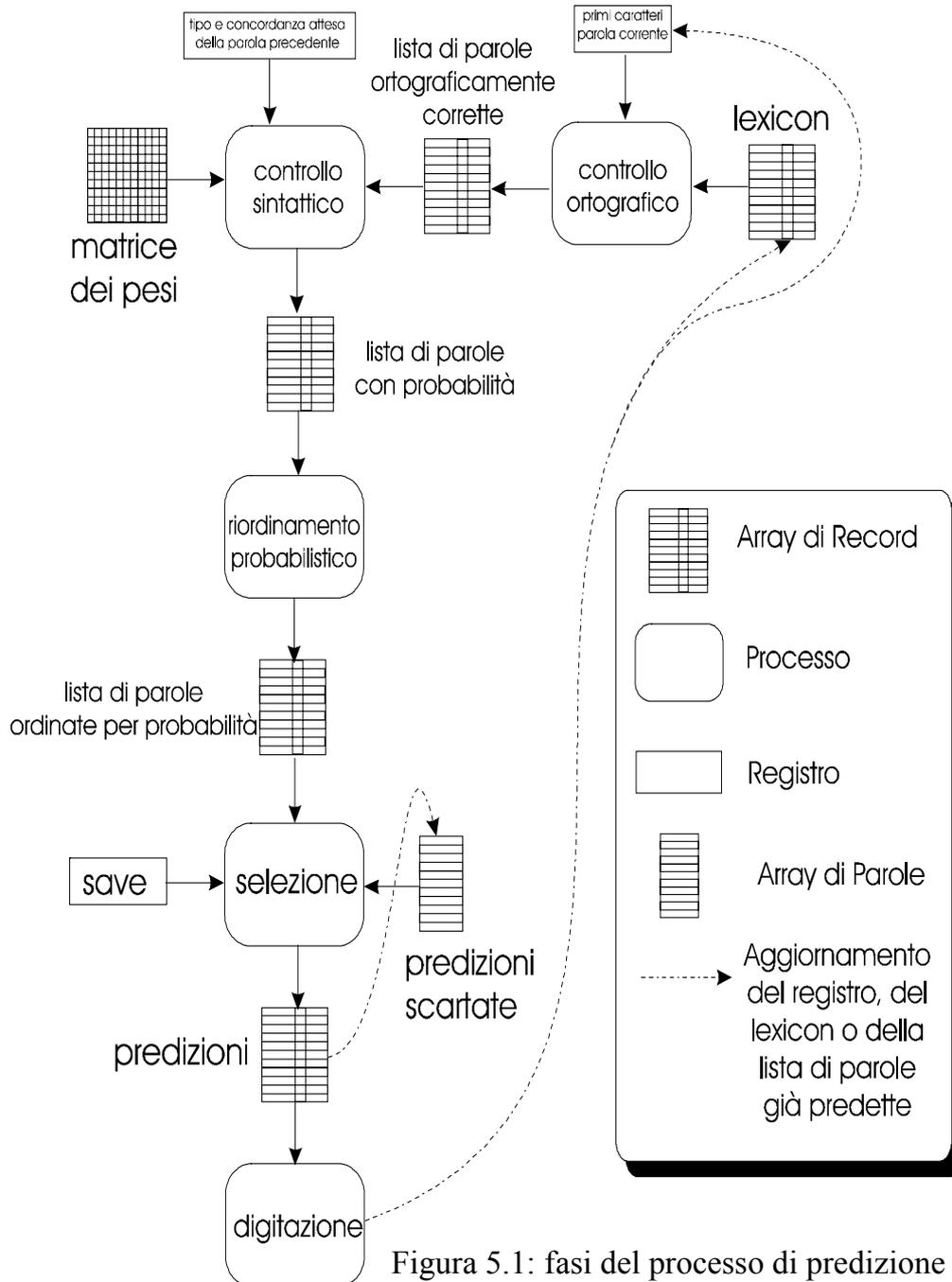


Figura 5.1: fasi del processo di predizione

### 5.3 Funzioni e Procedure varie.

Le fasi del processo esaminate nel paragrafo precedente vengono eseguite ogni volta che un carattere viene digitato, ma in determinate occasioni il programma si serve di altre funzioni e procedure:

- **inserisce:** Questa procedura si occupa di inserire una parola o frase nella finestra di editing nel punto in cui si trova il cursore. Quando necessario (ad esempio nell'espansione di acronimi) rimuove i caratteri già digitati per sostituirli con il risultato della predizione.
- **aggiorna lexicon:** Quando viene premuto lo spazio (o un altro carattere non alfabetico) il sistema conclude il ciclo di predizione ed esamina la parola digitata. La cerca quindi nel lexicon e ne incrementa il valore del campo `freq`; se non la trova, la aggiunge al dizionario. Quando nel lexicon la parola appena digitata è associata ad una espansione, la funzione opera automaticamente l'espansione servendosi della funzione *inserisce*.
- **aggiungi parole da file di testo:** Questa funzione utilizza come input un file di testo, per aggiornare il lexicon. Nella tabella 5.1 si vede che l'importanza dei metodi predittivi sintattici e statistici diventa sempre minore man mano che vengono digitati i caratteri della parola, quindi il lexicon deve contenere le parole che l'utente vuole usare. Servendosi di questa funzione l'utente può utilizzare qualsiasi suo documento da lui scritti per memorizzare il proprio stile, oppure creare dei lexicon specializzati in una certa area semantica.
- **aggiungi abb/espansione:** Questa funzione permette di aggiungere al lexicon acronimi che il programma espande automaticamente durante la scrittura. Questo accorgimento aumenta sia il `keystroke_saving` che il risparmio di tempo: durante l'uso della Word Prediction è necessario spostare continuamente lo sguardo dalla tastiera allo schermo, operazione che rallenta il processo di scrittura; l'espansione di sigle e abbreviazioni non richiede conferme e quindi non rallenta la digitazione. Non esiste uno standard per le abbreviazioni, quindi non sono definite per default come i lemmi e le frequenze del lexicon.

## CAPITOLO 5 - L' algoritmo di predizione

- ❑ **capitalizza:** automatizza la conversione in maiuscolo della prima lettera dopo un punto. Questa funzione non solo fa risparmiare un tasto (o due nel caso che si usi il tasto shift lock), ma è importante per chi può servirsi di una sola mano.
- ❑ **autospace:** inserisce automaticamente uno spazio dopo una parola o un segno di punteggiatura. Si tratta di un altro accorgimento utile per aumentare il *keystroke\_saving*.
- ❑ **selezione dizionario:** permette l'utente di scegliere quale lexicon viene utilizzato dal programma. Come si è visto nel paragrafo 3.3.4 implementazione della predizione semantica può essere ottenuta in modo approssimativo, utilizzando dizionari diversi in funzione dell'argomento del testo che si sta scrivendo. Tali dizionari possono essere agevolmente creati con la funzione *aggiungi parole da file di testo*.
- ❑ **non ripetere:** il programma non visualizza due volte lo stesso suggerimento durante la digitazione della parola. Questa funzione evita che il programma continui a suggerire una parola già scartata dall'utente (ad es. se alla digitazione della "b" è stato suggerito "bastone", dopo aver digitato "ba" il programma non suggerirà nuovamente "bastone"): permette sicuramente di migliorare il *keystroke\_saving* ma richiede una maggior attenzione da parte dell'utente.



## **6 Mind Reader: Text Editor con Word Prediction.**

Sulla base dell'algoritmo per la predizione del testo esposto nei capitoli precedenti, abbiamo realizzato il programma Mind Reader: un'applicazione che utilizza metodi di predizione alfabetici, sintattici e statistici adattivi per velocizzare la digitazione di un testo ASCII.

### **6.1 Obiettivi.**

La realizzazione del programma Mind Reader si propone di:

1. Sperimentare le prestazioni della word prediction al variare del metodo predittivo utilizzato.
2. Realizzare la parte centrale di un VOCA (Vocal Output Communication Aids), un ausilio comunicativo utilizzabile da una persona con disABILITÀ neuromotoria e con *speech impairment*.

Questi due obiettivi possono entrare in conflitto: infatti la facilità d'uso richiesta da un'applicazione di largo impiego è poco conciliabile con la complessità di un'interfaccia progettata a scopo di ricerca. Mentre nel primo caso si richiede un layout semplice, essenziale e per quanto possibile intuitivo, nel secondo è necessario che il programma fornisca un grande numero di informazioni sul processo in corso e sia dotato di numerose configurazioni modificabili durante l'esecuzione. Per ovviare a questo problema si è deciso di sdoppiare il layout del programma: mantenendo invariato il motore predittivo, sono state create due interfacce utente, una per la sperimentazione, un'altra per l'effettivo utilizzo.

### **6.2 Strumenti di sviluppo: scelte preliminari.**

Fino a non molto tempo fa realizzare ausili informatici per l'autonomia delle persone disABILI era proibitivo. Anche se le caratteristiche di versatilità e di adattabilità di tutti gli strumenti basati su un microprocessore sono notevoli, l'affidabilità, l'economicità e

l'ergonomicità dei sistemi informatici erano ancora insufficienti. Per di più un computer, per la sua difficoltà d'uso, veniva percepito più come un mezzo che separa e che allontana fisicamente, piuttosto che un mezzo per l'integrazione.

Le cose oggi stanno decisamente cambiando, soprattutto grazie ai più recenti strumenti per lo sviluppo di software visuali, che facilitano enormemente la realizzazione di interfacce di semplice uso e particolarmente ergonomiche (**per normovedenti**), inoltre lo sviluppo e la diffusione dei computer portatili permettono la realizzazione di ausili facilmente trasportabili.

### 6.2.1 Ambiente di sviluppo.

Per quanto riguarda la scelta del sistema operativo è stato scelto Windows 95, perché permette la realizzazione di interfacce grafiche semplici ed ergonomiche per chi ha una disABILITÀ neuromotoria. Inoltre PC portatili e multimediali che montano questo sistema operativo sono molto diffusi e sono un'ottimo punto di partenza per la realizzazione di un VOCA [25].

Riguardo al linguaggio di programmazione, Visual Basic della Microsoft e Toolbook, come altri linguaggi event-driven, non sono stati presi in considerazione, non essendo adatti per l'implementazione di algoritmi in grado di manipolare grandi quantità di stringhe. Il Prolog sembrava essere una buona soluzione per applicazioni di tipo linguistico-computazionale, ma manca di rapidità nel gestire una grossa base di dati: questa caratteristica è molto importante per questo genere di applicazione.

Il C++ ed il Modula sono linguaggi molto efficienti e diffusi in ambito accademico ma non sono adatti, almeno nelle versioni in nostro possesso, alla realizzazione di interfacce grafiche.

Java possiede caratteristiche intermedie tra la duttilità del Prolog e la velocità del C++, ma si è rivelato troppo lento in applicazioni test.

Si è optato per il Delphi 3 della Borland, che è un linguaggio Pascal orientato agli oggetti [24]. Alle caratteristiche del noto Turbo Pascal, aggiunge efficienti funzioni per manipolare basi di dati. In questo ambiente è semplice e rapida la creazione di interfacce grafiche. Esso inoltre genera, come il C++ ed il Modula, codici macchina direttamente eseguibili dalla CPU e quindi molto più veloci dei codici generati da Java,

Visual Basic e Prolog, che richiedono un'ulteriore elaborazione da parte del microprocessore, prima dell'esecuzione.

### 6.2.2 Hardware.

Allo scopo di rendere eseguibile l'applicazione su sistemi di basso costo, abbiamo definito entry-level i sistemi con almeno 16Mb di RAM e processore Pentium.

Per sviluppare il software è stato usato un computer dotato di microprocessore 486 DX4 con 16 Mb di RAM: questo garantisce l'efficienza dell'applicazione su macchine di classe superiore.

### 6.3 L'implementazione.

L'architettura del sistema (figura 6.1) è costituita da tre moduli che interagiscono : il motore predittivo, le basi di dati (lexicon e matrice dei pesi) e l'interfaccia utente.

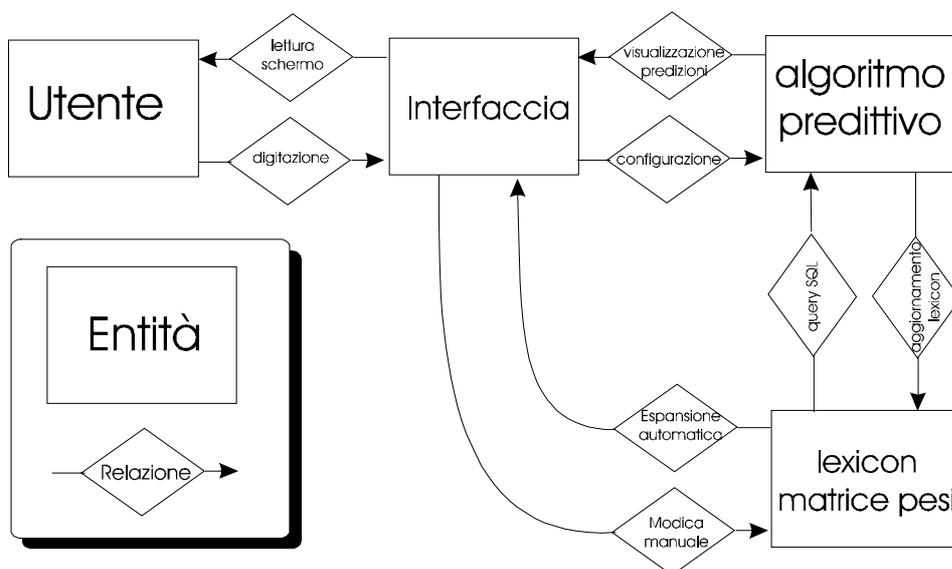


Figura 6.1: Architettura di Mind Reader

I problemi che si sono dovuti superare durante lo sviluppo dell'applicazione sono:

- **lexicon**: realizzare un vocabolario tipizzato per la lingua italiana sufficientemente ampio e articolato.
- **matrice dei pesi**: determinare valori per una efficiente predizione sintattica.
- **interfaccia**: creare un layout facile da usare.
- **motore**: realizzare un algoritmo efficiente, configurabile e in grado di sfruttare al massimo l'adattività del lexicon, per migliorare nel tempo la qualità delle prestazioni predittive fornite.

Analizziamo in dettaglio questi 4 punti.

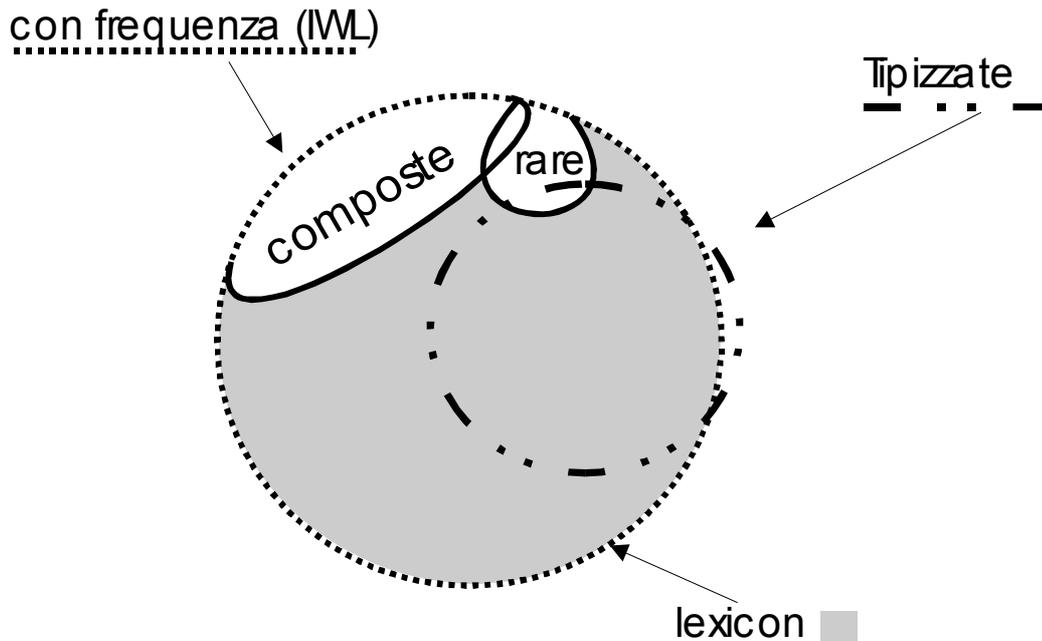
### 6.3.1 Lexicon

Le fonti utilizzate per la creazione del lexicon sono due liste di parole create dal gruppo del Prof. Delmonte dell'Istituto di Linguistica dell'Università di Venezia [16]. La prima (IWL: Italian Word List), è una lista di 30.000 lemmi della lingua italiana, ordinati per frequenza d'uso, ricavata dall'analisi automatica di un corpus di 500.000 occorrenze in testi italiani. La seconda contiene 28.000 parole classificate con diversi criteri, tra cui il tipo sintattico: molte si ripetono anche più volte, ma con etichette differenti. Di questi lemmi circa 10.000 sono univoci e con tipizzazione adatta ai nostri scopi. Unendo le due liste, alcune categorie sintattiche, ed eliminando le parole composte, quelle prive di classificazione per frequenza e quelle con frequenza uguale a 1 e quindi poco usate, si è ottenuto un elenco di 21119 lemmi (figura 6.1) ordinati per frequenza e parzialmente tipizzati, che costituisce il lexicon (*big\_lex*) di default del programma Mind Reader.

Il lexicon è organizzato come una tabella dBase 4, in cui ogni record è costituito da cinque campi (vedi tabella 5.2) che contengono nell'ordine la parola, la frequenza, il tipo sintattico, la concordanza attesa e l'eventuale espansione o sostituzione automatica.

Per velocizzare le funzioni di ricerca nella tabella è stato associato un indice che utilizza il campo *parola* come chiave: in questo modo è

possibile sfruttare la velocità e l'efficienza della ricerca binaria. Anche l'indice, come il lexicon, è adattivo poiché è in grado di tener conto sia dell'aggiunta che dell'eliminazione di record.



**Figura 6.2:** Struttura delle liste di parole e del lexicon *big\_lex*

Un secondo lexicon (*small\_lex*) è stato creato per svolgere un test più accurato sul sistema di predizione. Dalle 2500 parole più frequenti di *big\_lex* è stato ricavato, aggiungendo tutte le forme flesse mancanti degli aggettivi e dei sostantivi, un dizionario contenente circa 3000 parole con frequenza, suddivise in 23 diversi tipi sintattici e classificate secondo il genere ed il numero. Per ogni aggettivo e sostantivo sono state inoltre aggiunte tutte le forme singolare, plurale, maschile e femminile.

### 6.3.2 Determinazione automatica della matrice dei pesi

Nel paragrafo 5.1.2 sono stati descritti due sistemi per determinare i valori della matrice dei pesi: uno basato su modelli di frase e uno sull'analisi statistica di testi. Mind Reader può servirsi di entrambi i metodi: la matrice dei pesi può essere introdotta manualmente dall'utente o costruita in modo automatico, tramite un'apposita funzione implementata nel programma, dall'analisi statistica di un file di testo

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

(preferibilmente lungo). La procedura di analisi statistica è descritta dal seguente codice pseudo-pascal:

### Procedure DeterminaPesi

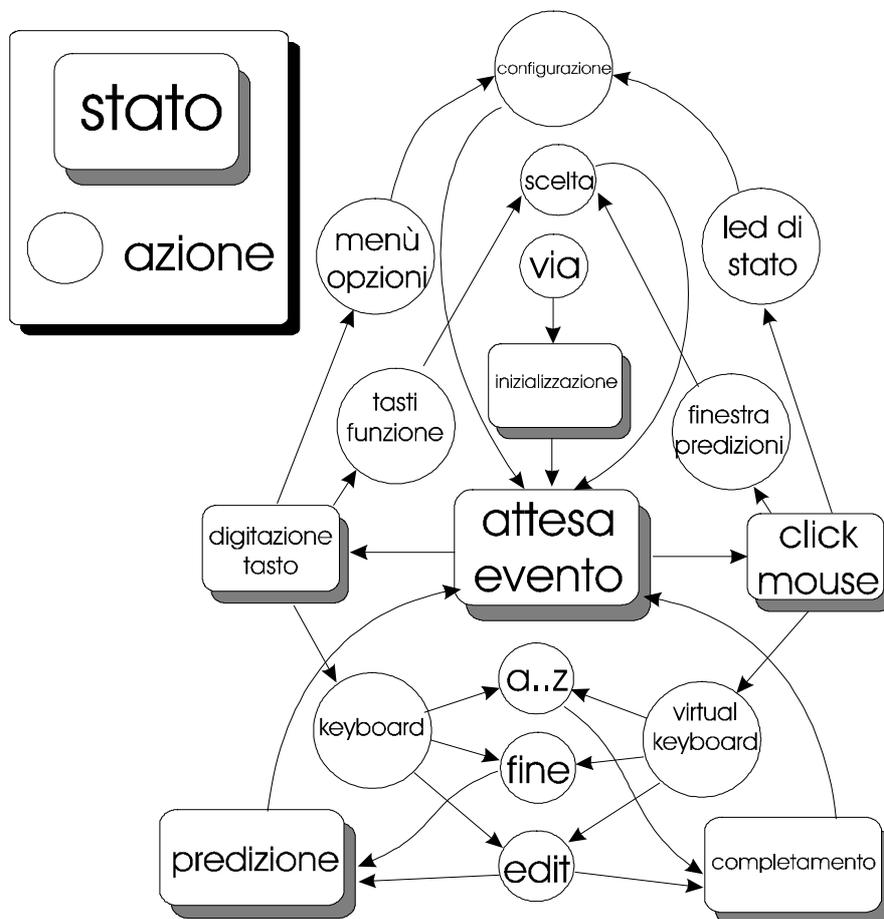
```
BEGIN
  TipoPrec:='other';           //la var TipoPrec viene inizializzata
  WHILE NOT eof(file.txt) DO   //cicla fino alla fine del file di input
    BEGIN
      read(file.txt, parola);   //dal file di input si legge una parola

      IF Lexicon.findkey([parola]) THEN //se la ricerca indicizzata della
        //se la ricerca indicizzata della
        //parola nel lexicon ha successo, il
        //puntatore si sposta a quel record
        //nel lexicon
        BEGIN
          inc(pesi[lexicon.tipo, TipoPrec]); //viene incrementato il valore
          TipoPrec:=lexicon.tipo           //corrispondente della matrice
        END                               //ed aggiornato il registro TipoPrec

      ELSE TipoPrec:='other';       //altrimenti pone TipoPrec='other'
    END;
  END;
```

### 6.3.3 Interfaccia

Gli studi compiuti sull'efficacia della word prediction indicano che questa è inversamente correlata al carico cognitivo [17]. Nel design del layout è quindi necessario minimizzare la quantità delle informazioni visuali, di cui l'utente si deve servire per usare la predizione, e renderne massima la leggibilità.



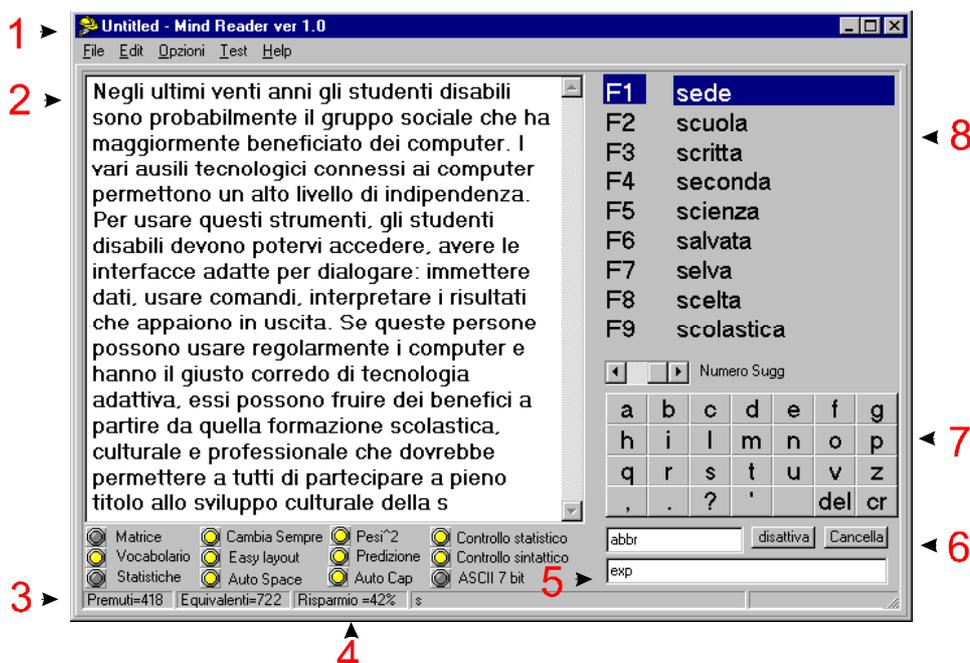
**Figura 6.3:** diagramma duale (keyboard/mouse) dell'interfaccia

C'è però il rischio di impoverire ed ostacolare la generazione e visualizzazione di quelle informazioni necessarie per analizzare le

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

prestazioni del sistema ed eventualmente variarne le impostazioni. Si rende quindi necessario lo sviluppo di due layout diversi che utilizzano lo stesso motore: uno più semplice e meno informativo adatto ad uso normale (**easy layout**), ed uno più ricco e complesso utilizzabile per sperimentare e testare le varie configurazioni (**test layout**).

Secondo le linee guida di Universal Design suggerite dal Trace Center dell'Università del Minesota [18], che caratterizzano i prodotti progettati in modo da aumentarne l'accessibilità, l'interfaccia è stata concepita con un'architettura duale (figura 6.3): ogni funzione del programma è attivabile sia attraverso il mouse che con la tastiera. In questo modo viene facilitato l'utilizzo del programma tramite dispositivi alternativi di input.



**Figura 6.4:** layout semplice di Mind Reader

In figura 6.4 è riprodotto l'**easy layout**: la descrizione dettagliata degli elementi di cui si compone è riportata nella Guida del programma (Help).

Le componenti principali dell'interfaccia sono:

- 1) **Barra dei menu a tendina**: contiene i comandi standard delle applicazioni Windows, le impostazioni configurabili dall'utente, le

funzioni di test, di aggiornamento del dizionario e della matrice dei pesi. In particolare la tendina *Opzioni* contiene il comando per selezionare il dizionario da utilizzare e una serie di opzioni binarie che consentono di variare i parametri di funzionamento del programma. E' possibile attivare o disattivare sia l'intero motore di predizione che i soli gli algoritmi statistici o sintattici, la ripetizione dei suggerimenti, la funzione di autospaziatura e automaiuscolo, l'adattività del vocabolario o delle statistiche. Infine si può scegliere se utilizzare il codice ASCII a 7 o 8 bit e il tipo di layout (easy o test).

La tendina *test* contiene le funzioni per creare o cambiare la matrice dei pesi, per aggiornare o addestrare automaticamente il vocabolario (*aggiungi parole da file di testo*, vedi paragrafo 5.3) e per eseguire i test sul sistema (vedi capitolo 7).

- 2) **Finestra di editing**: appare il testo digitato (colore e tipo dei caratteri sono selezionabili attraverso le voci *colori* e *caratteri* del menu *opzioni*).
- 3) **Barra di stato**: riporta alcune informazioni sul processo di predizione:
  - il numero di tasti premuti
  - il numero di tasti che si sarebbero dovuti premere
  - il *keystroke\_saving*
  - la parola in corso di digitazione
- 4) **Led**: componenti visuali con una duplice funzione: informano sullo stato delle opzioni binarie della tendina *Opzioni* e permettono di variarle.
- 5) **Impostazione dell'espansione/sostituzione**: due caselle di input in cui digitare l'abbreviazione e la parola corrispondente e un tasto per disattivare o attivare questa funzione.
- 6) **Cancella**: tasto per eliminare dal lexicon il record dell'ultima parola digitata.
- 7) **Tastiera virtuale**: permette a quegli utenti che non possono utilizzare la tastiera di digitare i caratteri tramite il mouse o un emulatore.

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

- 8) **Lista delle predizioni:** elenco dei suggerimenti di cui l'utente si può servire utilizzando i tasti funzione. Una barra a scorrimento permette di impostare il numero di suggerimenti prodotti dal programma.

Agli elementi visti, se ne aggiungono altri 4 quando viene selezionato il **test layout** (figura 6.5):

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

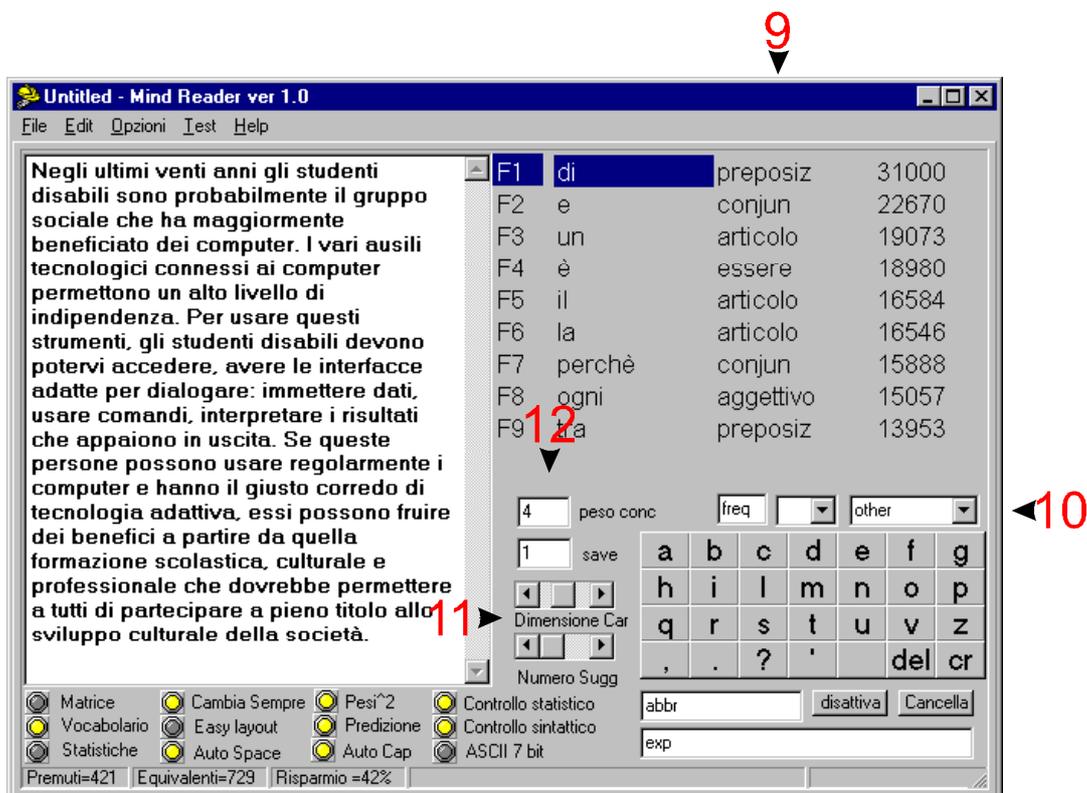


Figura 6.5: test layout di Mind Reader

- 9) **Lista delle predizioni aumentata:** aumenta la quantità di informazioni visualizzate; oltre alla parola viene anche indicato il suo tipo sintattico e la frequenza.
- 10) **Caselle di editing della parola:** visualizzano il contenuto dei campi frequenza, concordanza attesa e tipo sintattico dell'ultima parola digitata e ne consentono la modifica manuale.
- 11) **Barre di scorrimento:** *Dimensioni Caratteri* imposta la grandezza dei caratteri della lista dei suggerimenti, *Risparmio Minimo* e *Numero Suggerimenti* seleziona il numero delle parole proposte dal sistema.

- 12) **Caselle di configurazione:** Visualizzano il valore dei registri *save*, il *keystroke saving* minimo delle parole suggerite (vedi paragrafo 5.1.3), e *K*, la variabile che assegna il peso alla concordanza (vedi paragrafo 5.2.3) permettono di modificarli.

### 6.3.4 Funzione di Predizione.

L'ambiente di sviluppo scelto, Delphi 3, permette di realizzare un'interfaccia utente estremamente ergonomica ed informativa ma anche l'efficiente implementazione degli algoritmi di predizione illustrati nei capitoli precedenti.

La parte più importante del programma sono le operazioni di ricerca e modifica nel lexicon: la ricerca di un determinato lemma serve ad ottenere le informazioni ad esso correlate; l'estrazione di tutte le parole che iniziano in un certo modo permette la creazione del sublexicon; la possibilità di aggiungere, modificare e cancellare i record consente di avere un vocabolario che si adatta con l'uso alle caratteristiche dell'utente. E' importante che queste operazioni siano efficienti e veloci. La gestione del lexicon potrebbe essere resa più rapida caricandolo completamente nella RAM: in questo modo le operazioni di "fetch" sarebbero molto più rapide, ma il sistema diventerebbe vulnerabile in seguito ad improvvisi malfunzionamenti, che causerebbero la perdita delle informazioni acquisite durante la sessione in corso. Inoltre la mole del lexicon può essere notevole (alcuni Mb), fattore che crea un problema nei sistemi meno potenti.

Nel Delphi 3.0 sono implementate alcune istruzioni SQL (Structured Query Language) per interrogare basi di dati: tali funzioni sono molto efficienti e sicure. La soluzione adottata consiste nel servirsi di tali funzioni per le operazioni di selezione del lexicon (Query). Il risultato di questa operazione (sublexicon) viene quindi copiato in un array RAM per velocizzare le successive operazioni di ordinamento ed elaborazione .

L'interrogazione (query) SQL effettuata sul lexicon è:

```
SELECT * //seleziona tutti i campi...
FROM lexicon.dbf //...della tabella lexicon.dbf
WHERE parola LIKE ParCur // e tutti i record per cui il campo
parola inizia con gli stessi
caratteri registrati da ParCur
```

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

```
ORDER BY freq DESC           //ed ordinali in modo decrescente
                              usando il campo freq come
                              chiave
```

Successivamente il sistema ricopia il risultato della query in un array di record definito in questo modo:

```
MioTipoArray = ARRAY[1..n] OF voce
voce = RECORD
  parola : string[30]
  freq   : integer
  prob   : integer // =freq*peso[TipoPrec,tipo]*K
  tipo   : string[4]
  ConcAtt : string[1]
```

durante la copiatura il programma rielabora le frequenze (*freq*) secondo criteri sintattici, definendo così le probabilità (*prob*). A questo punto si può riordinare in funzione della probabilità l'array.

I primi `MaxSugg` elementi dell'array di record *sublexicon*, che hanno nel campo parola una stringa sufficientemente lunga e che non sono già stati proposti all'utente, vengono visualizzati.

Il programma rimane quindi in attesa di un evento, cioè della pressione di un tasto: un tasto lettera fornisce al sistema un'informazione per raffinare ulteriormente le proposte, un tasto funzione indica l'intenzione di usare una delle parole suggerite, un altro tasto (spazio, virgola, punto, ecc.) informa il sistema della fine della parola e attiva l'adattamento del lexicon. Se è stata digitata una parola nuova, questa viene salvata con una frequenza iniziale pari a 2 (il minimo valore di frequenza delle parole

$$freq = freq + p * \ln(freq + 1)$$

contenute nel lexicon) e tipo sintattico indefinito (*other*), altrimenti viene aggiornata la frequenza applicando una funzione logaritmica (in modo tale che i valori più alti di frequenza abbiano una minor tendenza a modificarsi):

Il moltiplicatore *p* determina l'aumento della frequenza; dopo alcune prove abbiamo scelto un valore  $p = 50$  nei nostri test. L'utente può

## CAPITOLO 6 - Mind Reader: Text Editor con Word Prediction

scegliere di disattivare il controllo sintattico o quello statistico (in questo caso le parole vengono presentate in ordine alfabetico). Se il controllo sintattico non è attivo il valore assunto dal campo `prob` è uguale a quello del campo `freq`.



## **7 Valutazione dell'efficacia di Mind Reader.**

La progettazione dell'interfaccia, questione che viene spesso trascurata o trattata in modo superficiale, è uno degli aspetti più importanti nel determinare il successo o il fallimento di un'applicazione informatica. Questo discorso è ancora più vero per i prodotti di Assistive Technology (AT) progettati principalmente per persone disABILI.

In questi anni le cose stanno cambiando: è diventato sempre più difficile stabilire le caratteristiche dell'utente-consumatore medio. Contemporaneamente le moderne tecnologie informatiche rendono più facile adattare un prodotto tecnologico alle necessità della persona e non viceversa.

In questo nuovo scenario sta assumendo una notevole importanza il concetto di **usability** [19]. Questo termine è stato definito da J. Preece come *"...la misura della semplicità con cui un sistema può essere appreso e usato, della sua sicurezza, efficacia, efficienza e dell'atteggiamento dell'utente nei confronti di questo"* [20].

Alla luce di questi concetti abbiamo eseguito dei test per valutare l'usability di Mind Reader. Le procedure utilizzate verranno descritte e commentate nel presente capitolo.

### **7.1 Criteri di valutazione dell'Usability.**

La definizione di Preece è molto elegante ma va dettagliata se la si vuole utilizzare come base per individuare dei test quantitativi.

#### **7.1.1 Semplicità dell'uso e dell'apprendimento.**

In molte situazioni l'utente ha urgente necessità di usare l'applicazione per comunicare, ma ha poco tempo per familiarizzarsi con il mezzo e non può rivolgersi a qualcuno in grado di istruirlo: in questi casi la semplicità d'uso e di apprendimento sono fattori determinanti affinché l'ausilio possa essere effettivamente utilizzato.

### **7.1.2 Sicurezza.**

L'incubo maggiore di chi usa ausili per la CMC è la perdita improvvisa dei dati, dovuta ad un malfunzionamento del sistema o dell'applicazione. Realizzare applicazioni ad alto livello di affidabilità richiede uno sforzo notevole a livello di codifica. Una strada alternativa per garantire la conservazione delle informazioni statistiche consiste nell'aggiornare continuamente il file del lexicon conservato nella memoria non volatile (disco fisso).

### **7.1.3 Efficacia.**

E' la misura di quanto bene il prodotto è in grado di svolgere il compito per cui è stato sviluppato. Questo fattore può essere valutato sia quantitativamente, attraverso parametri come il *keystroke\_saving*, che qualitativamente analizzando le *impressioni dell'utente*.

### **7.1.4 Efficienza.**

E' una misura legata allo sforzo richiesto per ottenere un certo livello di efficacia. E' possibile che un ausilio sia efficace perché in grado di compiere una determinata operazione, ma che lo sforzo cognitivo o il tempo richiesti lo rendano inutilizzabile. Ad esempio la scansione (vedi figura 3.1) è un metodo alternativo di input in cui il sistema propone in successione tutti i caratteri all'utente, che può selezionare quello desiderato servendosi di un unico tasto: si tratta di una soluzione efficace, ma talmente lenta da essere raramente efficiente per rendere la CMC accessibile a chi ha una grave disABILITÀ neuromotoria.

### **7.1.5 Motivazione.**

Il successo di ogni intervento riabilitativo, quale l'addestramento di una persona con disABILITÀ all'uso autonomo di prodotto di Assistive Technology, dipende in gran parte dalla voglia che ha l'individuo di accrescere le proprie opportunità partecipative. Questo fattore, seppur basilare, non può essere influenzato dal progettista. Una applicazione come Mind Reader, apre grandi possibilità di relazione a chi è affetto da un deficit comunicativo: dà infatti la possibilità di accedere in modo autonomo a mezzi di comunicazione, come la posta elettronica, in cui le differenze fisiche tra le persone non sono importanti [5].

## 7.2 Metodi e strumenti per la valutazione.

Per verificare l'efficacia e l'efficienza dei vari metodi predittivi implementati in Mind Reader (ortografico + statistico + sintattico), sono stati utilizzati tre parametri quantitativi.

- **Keystroke\_Saving**: (già definito nel paragrafo 3.1) misura l'efficienza del programma come risparmio di tasti premuti secondo la formula:

$$keystroke\_saving = \left( 1 - \frac{tasti\_premuti}{caratteri\_visualizzati} \right) * 100$$

- **Type\_Rate**: il numero di tasti premuti dall'utente ogni secondo; permette di valutare l'influenza del carico cognitivo indotto dal programma (la necessità di esaminare continuamente la lista dei suggerimenti).
- **Time\_Saving**: risparmio di tempo nella battitura di un testo da parte dell'utente; consente di misurare l'efficacia del programma.

Nel programma Mind Reader sono implementate due funzioni che permettono di calcolare automaticamente tutti questi parametri.

Mentre il *Keystroke\_Saving* dipende solamente dal programma e ne misura le prestazioni teoriche (la capacità di effettuare predizioni corrette), gli altri due parametri forniscono fondamentali informazioni sulla reale utilità del sistema, ma dipendono fortemente dalle caratteristiche dell'utente. Uno studio preciso richiederebbe quindi l'esecuzione di test su un largo numero di utenti (test di questo tipo sono già in progetto). In questo stadio preliminare le prove condotte da un singolo utente "medio" forniscono utili indicazioni.

### 7.2.1 Misura dell'efficacia: *Keystroke\_Saving*.

Il calcolo del *keystroke\_saving* su un testo assegnato, contenuto in un file (file.txt), è compito di un'apposita procedura contenuta nel programma, in grado di simulare la digitazione del testo e di salvare su un file (dati.txt) la lunghezza di ogni parola digitata e quanti tasti sarebbe stato sufficiente premere con la Wpd. Da queste due informazioni è possibile calcolare il *keystroke\_saving* locale (parola per parola) e quindi quello globale (l'intero testo).

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

Vediamo questa procedura, descritta in pseudo-pascal

### **Procedure SimulaDigitazione**

```
BEGIN
WHILE NOT (eof(file.txt)) DO // cicla fino alla fine del file.txt
  BEGIN
  n:=0;
  read(file.txt, parola); //dal file.txt legge una parola
  REPEAT
  trovata:=FALSE
  sendmessage(edit.handle, parola[n]); //simula la pressione di un tasto
  predict(ParCur, MaxSugg, ConcAtt, TipoPrec, Save); //genera MaxSugg predizioni
  IF parola=predizione[m] THEN //se fra le m predizioni generate da
  BEGIN //predict() c'è quella giusta...
    writeln(dati.txt, length[parola]+' `'+n); //scrive nel file di out i dati
    //per il calcolo del keys_saving
    trovata:=TRUE
  END;
  inc(n)
  UNTIL trovata
  END;
END
```

Al di là del caso di un testo memorizzato a priori, Mind Reader misura durante ogni digitazione il numero dei tasti premuti, quello dei caratteri visualizzati e calcola il *keystoke\_saving* ottenuto fino a quel momento (tutti questi dati vengono visualizzati dall'interfaccia, vedi la voce *barra di stato* nel paragrafo 6.3.3)

### **7.2.2 Misura dell'efficienza: *type\_rate* e *time\_saving* .**

La determinazione del *type\_rate* e del *time\_saving* viene effettuata dal programma mediante una procedura che salva ad intervalli di tempo regolari il valore assunto dalla variabile *ParCur* (paragrafo 5.1.3): in tal modo è possibile misurare il tempo necessario alla battitura delle singole parole e dell'intero testo. Da questo dato, conoscendo il numero di tasti premuti, si calcola il *type\_rate*; il *time\_saving* si calcola invece conoscendo il tempo impiegato per battere il testo senza predizione.

### 7.3 Ottimizzazione del programma.

I test di valutazione del Mind Reader sono stati eseguiti utilizzando il brano riportato in figura 7.1., tratto da *Guida al consumo critico*, EMI, Bologna 1996, pag. 31, preso come esempio di testo “generale”, non tecnico.

Il lexicon utilizzato durante i primi test è un'estensione di *small\_lex* (cfr. 6.3.1), che contiene circa 3000 parole ed è completamente tipizzato. Al dizionario sono state aggiunte le parole del brano che non vi erano già contenute: dei nuovi verbi sono state aggiunte anche le principali forme coniugate, degli aggettivi e dei sostantivi tutte le forme flesse.

Ora dobbiamo scegliere. Il mondo sta correndo un grosso rischio. Se vogliamo sostenere il pericolo dell'ingiustizia e della distruzione del pianeta, continuiamo a consumare come abbiamo fatto finora. Ma se vogliamo salvare il pianeta, se vogliamo cambiare le cose, allora dobbiamo consumare meno e dobbiamo prendere le distanze dalle imprese che si comportano ingiustamente. Noi possiamo indurre le imprese a fare scelte diverse. Quando facciamo la spesa ci dobbiamo ricordare che siamo potenti e che le imprese dipendono da noi. Il consumo critico funziona, l'esperienza ha dimostrato che quando i consumatori hanno fatto sentire la loro voce, le imprese hanno cambiato perchè non volevano perdere la quota di mercato. Da questa esperienza possiamo trarre una seconda conclusione: quando la strada non esiste la possiamo costruire da soli. Del resto il commercio equo e solidale ha confermato questa possibilità. Il principio di fondo è di garantire ai produttori un giusto compenso per il loro lavoro. Nel commercio normale questo obiettivo è ostacolato dalla presenza degli intermediari. La soluzione è quella di distribuire direttamente ai consumatori i prodotti comprati dai contadini e dagli artigiani, perchè possano godere di tutto il prezzo pagato. Questa idea è nata in Olanda trenta anni fa, da allora si sta diffondendo in tutto il mondo.

**Figura 7.1:** Brano di prova usato per i test (circa 200 parole).

Il primo passo nella sperimentazione del programma consiste nella determinazione della configurazione ottimale. Due sono i parametri che influenzano la predizione sintattica: la matrice dei pesi e la concordanza.

### 7.3.1 Determinazione della matrice dei pesi.

Abbiamo inizialmente impiegato due matrici dei pesi. La prima (adhoc.pesi) è stata determinata automaticamente da Mind Reader applicando al brano di prova la funzione descritta nel paragrafo 6.3.2.

Questa matrice, per il modo in cui è stata determinata, dovrebbe fornire ottime prestazioni per il brano di prova, ma è stata calcolata conoscendo già il testo da battere e quindi viene utilizzata solo come termine di confronto.

La seconda matrice (norm.pesi) è stata determinata in modo euristico usando un metodo analogo a quello descritto nel paragrafo 5.1.2; scegliendo il modello di frase riportato in appendice B. Abbiamo assegnato il valore 3 alle combinazioni più frequenti (articolo-nome, verbo-avverbio, verbo stare-gerundio), il valore 2 a quelle usate ma meno frequenti (verbo stare-pronome, pronome-aggettivo) e rispettivamente i valori 1 e 0 a quelle rare o impossibili (articolo-articolo, verbo stare-verbo essere).

La terza matrice (norm<sup>2</sup>.pesi) e la quarta (norm<sup>3</sup>.pesi) sono ricavate da norm.pesi elevando rispettivamente alla seconda e alla terza ogni singolo peso. In questo modo è possibile aumentare l'influenza dell'algoritmo sintattico e valutare gli effetti di questi cambiamenti sull'efficacia della predizione.

L'efficacia di queste matrici è stata determinata eseguendo i test con il solo algoritmo sintattico. In questi esperimenti abbiamo posto  $K=5$  (cfr. 5.2.3). I risultati sono illustrati nella tabella 7.1, in cui sono riportati, in funzione del numero massimo di suggerimenti proposti, i valori di *keystroke\_saving* ottenuti tramite la procedura *Simula Digitazione*, usando le diverse matrici.

**Tabella 7.1:** *Keystroke\_saving* ottenuto applicando il solo algoritmo sintattico<sup>a</sup> di Mind Reader al testo di figura 7.1, variando il numero massimo di suggerimenti e la matrice dei pesi.

MaxSugg	Norm	norm <sup>2</sup>	norm <sup>3</sup>	Adhoc
1	33%	34%	34%	35%
2	39%	40%	40%	40%
3	42%	42%	42%	43%
4	43%	44%	44%	45%
5	43%	44%	44%	45%
6	44%	44%	44%	46%
7	44%	45%	45%	46%
8	45%	45%	45%	47%
9	45%	46%	46%	47%

a) Per tutti i test  $K = 5$  (vedi testo).

Analizzando la tabella 7.1, si vede che la matrice di pesi ottimale è *adhoc.pesi*. Le altre matrici forniscono valori molto vicini a quelli ottenuti con la matrice *adhoc*: questo conferma la validità del procedimento euristico utilizzato per determinarle. Da notare che elevare al quadrato i valori dei pesi migliora i risultati, ma utilizzare potenze maggiori non porta vantaggi.

Elevare a potenza (con esponente  $> 1$ ) i pesi significa aumentare la probabilità associata alle parole appartenenti ad un tipo plausibile: questo aumenta l'efficacia della predizione. Sulla base di questi dati si è scelto di utilizzare la matrice *norm<sup>2.pesi</sup>* nei test successivi.

### 7.3.2 Ottimizzazione della concordanza.

Il secondo passo da compiere per ottimizzare le prestazioni del programma è valutare l'influenza del parametro  $K$ . Nei primi test abbiamo posto  $K=5$ . Per verificare la bontà di questa scelta sono stati effettuati dei controlli variando il valore di  $K$  da 1 a 5: i risultati sono riportati in tabella 7.2.

**Tabella 7.2:** *Keystroke\_saving* al variare del coefficiente moltiplicativo  $K^a$ .

K	1	2	3	4	5
Keystroke_saving	42%	44%	44%	44%	44%

a) Matrice dei pesi:  $\text{norm}^2$ .pesi; Numero di suggerimenti ( $\text{MaxSugg}$ ) = 4; algoritmo di predizione: sintattico.

La tabella 7.2 conferma che aumentare il parametro di concordanza da 1 a 2 migliora i risultati; non si ottengono incrementi per  $K > 2$ . Per mantenere l'omogeneità con i test già effettuati, abbiamo deciso di continuare a utilizzare il valore  $K = 5$ .

## 7.4 Risultati dei test.

Determinata la configurazione ottimale di tutti i parametri, è possibile passare alla valutazione dell'efficacia dei vari algoritmi predittivi e dell'efficienza dell'applicazione.

### 7.4.1 Efficacia.

La misura dell'efficacia dei diversi algoritmi predittivi implementati in Mind Reader è stata effettuata valutando, tramite la procedura *simula digitazione*, il *keystroke\_saving* ottenuto con diverse combinazioni dei vari metodi. I risultati sono riportati in tabella 7.3. La prima colonna di valori è stata ottenuta attivando sia il controllo statistico che quello sintattico. Nella seconda e nella terza sono riportate le prestazioni del solo controllo statistico o di quello sintattico, rispettivamente. Infine nel quarto caso la lista dei suggerimenti è stata generata utilizzando come unico criterio l'ordine alfabetico.

**Tabella 7.3:** *Keystroke\_saving* ottenuto al variare del numero massimo di suggerimenti e del tipo di predizione.

<b>MaxSugg</b>	<b>sint<sup>a</sup>+stat</b>	<b>stat</b>	<b>sint<sup>a</sup></b>	<b>Alfabetica</b>
<b>1</b>	33%	27%	34%	26%
<b>2</b>	37%	32%	40%	31%
<b>3</b>	41%	36%	42%	35%
<b>4</b>	43%	37%	44%	37%
<b>5</b>	43%	38%	44%	38%
<b>6</b>	44%	40%	44%	40%
<b>7</b>	46%	42%	45%	40%
<b>8</b>	47%	44%	45%	42%
<b>9</b>	47%	44%	46%	43%

a) Matrice dei pesi:  $\text{norm}^2.\text{pesi}$ ;  $K = 5$ .

La tabella 7.3 mostra che si ottengono buoni risultati sia con il solo ordinamento alfabetico, che con il solo ordinamento statistico, che con il solo metodo sintattico. L'algoritmo sintattico fornisce ottime prestazioni, addirittura migliori di quello statistico. L'uso combinato di questi due metodi porta ad un inaspettato peggioramento dell'efficacia.

Per comprendere meglio questi risultati è opportuno illustrare la struttura del lexicon utilizzato. Ogni sostantivo e aggettivo è memorizzato nelle sue diverse forme flesse (maschile, femminile, singolare e plurale) per lo più con frequenze eguali. Il metodo statistico è quindi costretto a sprecare gran parte dei suggerimenti a disposizione per parole scorrette sintatticamente; tale effetto negativo è molto evidente quando il numero massimo di predizioni è basso. Il metodo sintattico può invece determinare la scelta della parola da suggerire basandosi sulla concordanza di genere e numero e questo lo rende particolarmente efficace. Analogamente l'algoritmo sintattico è talvolta in grado di selezionare la più probabile fra le diverse forme coniugate di un verbo: ad

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

esempio dopo un verbo servile vengono suggeriti verbi all'infinito, dopo il verbo avere participi passati di verbi transitivi

E' interessante notare che l'aumento dell'efficacia della predizione è molto piccolo passando dall'ordinamento alfabetico a quello statistico. Questo dipende dal fatto che il lexicon utilizzato è molto piccolo e quindi il criterio di ordinamento scelto è poco importante; inoltre le parole del brano di prova aggiunte a *small\_lex* hanno un valore di frequenza basso e quindi vengono suggerite tardi.

**Tabella 7.4:** *keystroke\_saving* ottenuto ripetendo la digitazione del testo di prova con il lexicon *small\_lex* non aumentato.  $\text{MaxSugg} = 4$ .

Ciclo di battitura	stat+sint <sup>a</sup>	Stat	Sint <sup>a</sup>	Alfa
1	39%	34%	39%	32%
2	47%	45%	43%	39%
3	48%	48%	43%	39%

a) Matrice dei pesi:  $\text{norm}^2$ .pesi;  $K = 5$ .

L'efficacia della predizione statistica e dell'adattività emerge più chiaramente dai risultati della tabella 7.4, dove vengono riportati i risultati relativi a 3 battiture successive del testo in figura 7.1. Nella prima digitazione del testo è stato utilizzato il dizionario *Small\_lex* non addestrato. Durante la digitazione, Mind Reader aggiunge al lexicon le parole nuove ed aggiorna le frequenze delle altre. Dal secondo ciclo non ci sono più parole nuove, ma solo frequenze da aggiornare: per questo motivo la predizione sintattica e quella alfabetica non migliorano più passando dal secondo al terzo ciclo.

Analizzando i dati in tabella 7.4 si nota come il sistema sia efficace anche con un vocabolario non addestrato (anche in questo caso la predizione sintattica è la migliore). Alla seconda battitura aumenta l'addestramento, aumenta l'efficacia generale, ma soprattutto quella del metodo statistico, che supera quella del metodo sintattico. In queste condizioni il metodo migliore di predizione è quello che si serve sia dell' algoritmo statistico che di quello sintattico. Alla terza battitura la

differenza tra i metodi statistico e statistico+sintattico diviene nulla: le frequenze del vocabolario sono ormai completamente adattate al brano di prova e le ulteriori informazioni provenienti dall'analisi statistica divengono superflue.

Un ulteriore esempio dell'importanza dei metodi statistici e dei limiti di quelli sintattici ci viene dai test condotti utilizzando il dizionario *big\_lex*: questo dizionario è molto più esteso (20.000 parole) ma caratterizzato nei tipi in maniera parziale e più grossolana e con concordanze quasi inesistenti. I risultati ottenuti sono riportati in tabella 7.5.

**Tabella 7.5:** Copertura del vocabolario (*hit\_rate*) e *keystroke\_saving* al variare del metodo predittivo e del vocabolario usato. MaxSugg = 4.

	Hit_rate	Sint/stat	Stat	Alfabetico
Small_lex	75%	39%	34%	32%
Big_lex	87%	41%	41%	29%

Un nuovo parametro che informa sull'efficacia dei programmi di WPD è la copertura del vocabolario (*hit\_rate*): la percentuale di parole del brano già presenti nel lexicon alla prima digitazione. Si nota immediatamente che l'utilizzo di *big\_lex* provoca un sostanziale aumento di questo parametro. Il programma conosce quindi la maggior parte delle parole del testo, ma deve cercare fra un numero di parole molto più alto che in *small\_lex*: questo fa sì che il metodo alfabetico diventi meno efficace, mentre quello statistico migliora notevolmente, fino a diventare più efficiente persino di quello statistico+sintattico, basato sullo *small\_lex*. Il numero di parole conosciuto dal programma è quindi più importante dell'utilizzo di un algoritmo sintattico. Usando *big\_lex* non si ha alcun miglioramento utilizzando anche il metodo sintattico: questo comportamento è dovuto alla scarsa tipizzazione di questo lexicon, che rende superfluo ogni tentativo di controllo sintattico.

In tabella 7.6 vediamo che l'efficacia della predizione statistica basata sul dizionario *big\_lex* migliora con le battiture, come ci si aspetta. La

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

conoscenza di tutte le parole del testo (a partire dal secondo ciclo) porta ad un notevole miglioramento delle prestazioni.

**Tabella 7.6:** *Keystroke\_saving* ottenuto usando la predizione statistica con il lexicon *big\_lex*, ripetendo la digitazione del testo e variando MaxSugg.

Ciclo	MaxSugg					
	1	2	3	4	5	6
1	30%	33%	39%	41%	43%	44%
2	38%	43%	47%	49%	50%	50%
3	42%	45%	51%	52%	52%	53%
4	44%	46%	52%	53%	53%	54%
5	45%	47%	52%	53%	54%	54%
6	45%	47%	52%	53%	54%	54%

In definitiva, i test effettuati indicano che l'algoritmo di predizione sintattica permette di aumentare l'efficacia di un programma di Word Prediction, ma solo se è associato ad un lexicon completamente classificato per tipo, genere e numero. Le dimensioni del lexicon utilizzato sono altresì molto importanti: si ottengono migliori risultati utilizzando un lexicon grande non tipizzato che uno piccolo, ma ben classificato.

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

Un'ultima serie di test ha riguardato l'efficacia della funzione che permette di non suggerire nuovamente le parole già proposte nello stesso ciclo di predizione (si veda paragrafo 5.3). Come si ricava dai dati riportati in tabella 7.7, questo accorgimento permette in genere di aumentare di due o tre unità percentuali il *keystroke\_saving* quando il numero di predizioni visualizzate è basso. L'utilizzo di questa funzione comporta un aumento del carico cognitivo per l'utente, che deve controllare attentamente la lista dei suggerimenti dopo ogni carattere digitato, ma come vedremo nel paragrafo seguente, in certi casi permettere di incrementare il *time\_saving*.

**Tabella 7.7:** *Keystroke\_saving* ottenuto al variare del numero massimo di suggerimenti (MaxSugg) e del tipo di predizione utilizzando la funzione di non ripetizione<sup>a</sup>.

MaxSugg	tipi di predizione			
	sint <sup>b</sup> +stat	statistico	sintattico <sup>b</sup>	alfabetico
1	36%	30%	37%	29%
3	43%	40%	44%	38%
6	45%	42%	46%	41%

a) Lexicon: *Small\_lex*.

b) Matrice dei pesi: norm<sup>2</sup>.pesi; K = 5.

### 7.4.2 Efficienza.

Alcuni test per la valutazione del *time\_saving* e del *type\_rate* sono stati effettuati il giorno 9/1/1998 da una studentessa affetta da una tetraparesi multipla utilizzando il brano, da lei scelto, presentato in figura 7.2:

*E' evidente che tutte le informazioni genetiche di cui dispone un animale non sono sufficienti per specificare tutte le interconnessioni neuronali che vengono stabilite e che probabilmente ammontano a un milione di miliardi. Perciò lo sviluppo del sistema nervoso deve avvenire anche attraverso processi epigenetici che attivano, nel corso dello sviluppo, gruppi specifici di geni secondo combinazioni appropriate in epoche particolari.*

**Figura 7.2:** Testo usato per verificare l'efficacia di Mind Reader

I risultati ottenuti sono riassunti in tabella 7.8. L'utente, che già aveva utilizzato programmi di text editing in ambiente Windows, si è familiarizzato velocemente con il programma. Sono state compiute quattro sessioni utilizzando il lexicon *big\_lex* e la sola predizione statistica: nella prima (*normale*) l'utente ha digitato il testo senza Word Prediction, nelle altre tre (*Wpd*) si è servita della Word Prediction.

Nelle prime due sessioni *Wpd* l'utente ha utilizzato il programma da sola, nella terza sessione *Wpd* era invece affiancata da una persona che le suggeriva il tasto funzione da premere quando il sistema forniva il suggerimento corretto. In questo modo l'utente non era costretto ad esaminare continuamente la lista di suggerimenti per cercarvi la parola voluta.

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

**Tabella 7.8:** Tabella riassuntiva dei test di valutazione dell'efficacia di Mind Reader<sup>a</sup> compiuti da una studente affetta da disABILITÀ motoria.

	normale	WPd 1°	WPd 2°	WPd 3 <sup>ob</sup>
Type_rate	0,50	0,30	0,30	0,40
Tasti premuti in totale	527	328	274	199
Caratteri visualizzati ogni s.	0,5	0,55	0,60	1,00
Tempo totale	17'30"	15'30"	15'06"	8'16"
<b>Time Saving</b>	0%	11%	19%	53%
<b>Keystroke Saving<sup>c</sup></b>	-0,2%	36%	47%	62%

a) Lexicon = *big\_lex*; MaxSugg = 5.

b) Con suggerimento verbale.

c) Il valore di *keystroke\_saving* riportato comprende gli errori di battitura: di qui il risultato negativo della prima colonna.

L'analisi dei risultati riportati in tabella 7.8 permette di fare le seguenti considerazioni:

- a) Il *keystroke\_saving* aumenta in maniera notevole con il susseguirsi delle sessioni: questo avviene certamente perché l'utente si familiarizza sempre più con il programma, imparando a sfruttarne le potenzialità in maniera migliore, ma anche perché il programma impara il testo utilizzato.
- b) All'aumento del *keystroke\_saving* corrisponde un aumento solo marginale del *time\_saving* nelle sessioni WPd 1 e 2: la necessità di esaminare la lista dei suggerimenti provoca una diminuzione del *type\_rate* (un rallentamento della velocità di battitura) che annulla quasi del tutto il vantaggio della diminuzione dei tasti premuti. Possiamo dire che non vi è un aumento del time-saving dovuto alla memorizzazione dei tasti da parte dell'utente.
- c) Eliminando la necessità di leggere la lista dei suggerimenti si ottiene un notevole aumento del *time\_saving*: questo risultato è dovuto in parte ad un miglioramento del *keystroke\_saving*

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

(imputabile sia all'aggiornamento del vocabolario che alla presenza del suggeritore, che permette l'utilizzo ottimale dei suggerimenti), ma soprattutto ad un incremento del *type\_rate*.

Il trasferimento dell'efficacia del sistema verso l'efficienza non è quindi immediato: in condizioni normali il tempo risparmiato è praticamente nullo, vi è un aumento del carico cognitivo e una diminuzione della quantità di movimenti (fattore importante nel caso di disABILITÀ neuromotorie). Questi risultati sono dovuti essenzialmente al fatto che l'utente viene costretto a leggere ed esaminare dopo ogni digitazione la lista dei suggerimenti.

Nella configurazione utilizzata, il programma risulta quindi essere un utile ausilio per la produzione di testi scritti (lettere, messaggi di posta elettronica), ma non per la comunicazione interpersonale in tempo reale. Per ottenere uno strumento efficiente anche in quest'ultima funzione è possibile lavorare in due direzioni:

- a) Aumentare il *keystroke\_saving*. Questo può essere ottenuto seguendo diverse strade: implementando la predizione sintattica su vocabolari ben tipizzati di grandi dimensioni, utilizzando la funzione di non ripetizione, e ottimizzando il programma rispetto al singolo utente per mezzo di dizionari personalizzati e dell'espansione di acronimi.
- b) Migliorare l'interfaccia tra programma e utente. Il programma può essere fornito di dispositivi alternativi di input o di output. Ad esempio l'utilizzo di un mouse a controllo ottico (eyetracking mouse) [21] permetterebbe all'utente con disABILITÀ neuromotoria grave di controllare contemporaneamente una tastiera virtuale (già implementata in Mind Reader) e la lista dei suggerimenti. Alcune prove preliminari suggeriscono la possibilità di utilizzare il riconoscimento vocale al posto della tastiera. Infine un feedback sonoro, associato ad una corta lista di suggerimenti, potrebbe permettere di digitare le parole senza spostare l'attenzione dalla tastiera: questa soluzione è sicuramente promettente, visto il buon *keystroke\_saving* ottenuto usando un solo suggerimento, lo *small\_lex* e la funzione di non ripetizione (tabella 7.7, riga 1).

Alcune delle soluzioni suggerite al punto (a) sono già implementate in Mind Reader ma non è stato possibile sottoporle a test per mancanza di tempo. Tali controlli sono comunque in programma.

## 7.5 Discussione dei risultati.

I test condotti permettono di valutare l'efficacia di Mind Reader in rapporto a quella dei programmi commerciali esistenti (vedi tabelle 7.9 e 7.10). Il confronto è puramente qualitativo poiché deve essere fatto con prodotti funzionanti per la lingua inglese [22], non essendo disponibili dati sui rari programmi sviluppati per trattare l'italiano (vedi appendice C). Mind Reader ottiene buoni risultati sia in termini di *keystroke\_saving* che riguardo alla copertura del vocabolario, ma i test effettuati sono stati compiuti su un unico brano di prova.

**Tabella 7.9:** Valutazione dell'efficienza di Mind Reader<sup>a</sup>

Programma	Keystroke_saving	Hit_Rate
Mind Reader ( <i>small_lex</i> )	39%	75%
Mind Reader ( <i>big_lex</i> )	41%	87%

a) vedi tabella 7.5

**Tabella 7.10:** Prestazioni di 5 programmi per la Word Prediction in inglese<sup>b</sup> [22].

Programma	Keystroke_saving	Hit_Rate
EZ Keys	45%	91%
Write 100	45%	45%
Predictive Linguistic Program	41%	86%
Word Strategy	36%	79%
GET	31%	84%

c) Ricavate dalla battitura di 20 brani in inglese di 500 parole, composti da persona con diverso grado di istruzione.

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

Le valutazioni compiute (tabella 7.4) permettono di esprimere primi giudizi sull'efficacia del metodo di predizione implementato: il semplice algoritmo di predizione sintattica debole utilizzato consente di migliorare le prestazioni. I risultati sono incoraggianti se confrontati con gli studi compiuti, sempre per la lingua inglese, sulla predizione sintattica forte, operante con CFG aumentate, del programma Windmill [11] (tabella 7.11). Entrambi i programmi ottengono un aumento del *keystroke\_saving* passando dall'ordinamento alfabetico alla predizione solo sintattica. E' importante ricordare che in italiano l'esistenza di molte forme flesse o coniugate per ogni radice verbale fa sì che, su lexicon di grandi dimensioni, gli ordinamenti alfabetici, ma anche statistici siano fortemente penalizzati.

**Tabella 7.11:** Efficacia della predizione sintattica nei programmi Windmill e Mind Reader. (MaxSugg= 5)

Programma	sintattico	alfabetico
Mind Reader <sup>a</sup>	45%	40%
Wind Mill <sup>b</sup>	37%	30%

a) Lexicon: *Small\_lex*. Matrice: norm<sup>2</sup>.pesi; K = 5.

b) Vocabolario di 1750 parole completamente tipizzato; test effettuati su tre brani in inglese di 500 parole.

Questo effetto è meno evidente con i piccoli vocabolari utilizzati per i test, ma diventerebbe importante con i lexicon di grandi dimensioni necessari negli usi pratici: in tali situazioni l'utilizzo di algoritmi sintattici efficienti potrebbe diventare di grande importanza. Lo svantaggio principale dei metodi sintattici è che richiedono la disponibilità di un lexicon tipizzato in modo completo e dettagliato. I test da noi effettuati dimostrano che una tipizzazione meno dettagliata non fornisce alcun risultato utile (tabella 7.5).

I test sull'efficacia confermano un fenomeno ben noto nella letteratura sulla Word Prediction [23]: quando l'utente possiede una velocità di battitura abbastanza buona (superiore alle 20 parole al minuto), l'uso di

## CAPITOLO 7 - Valutazione dell'efficacia di Mind Reader

questi ausili può portare ad un complessivo rallentamento della velocità di produzione del testo. In ogni caso i programmi di Word Prediction si rivelano molto efficaci per persone con gravi disABILITÀ neuromotorie perché consentono un minor numero di movimenti e una minor quantità di errori di battitura.

E' nondimeno ipotizzabile che una maggior efficacia possa essere ottenuta sia migliorando l'efficienza del programma e utilizzando opportune interfacce multimediali ridondanti ma anche addestrando l'utente a sfruttare in maniera ottimale tutte le potenzialità dell'applicazione.



## **8 Sunto e conclusioni.**

### **8.1 DisABILITÀ e CMC.**

Il progresso delle tecniche mediche permette la sopravvivenza e l'inserimento sociale di un numero sempre più alto di persone che, per malattia, incidente o vecchiaia, perdono qualche abilità.

Fra le tante patologie invalidanti sono particolarmente gravi quelle che colpiscono le normali abilità comunicative, rendendo difficile se non impossibile un coordinamento motorio sufficiente per controllare la parola e la scrittura. E' per ovviare a questo problema che vengono sviluppati e applicati sistemi per la Comunicazione Aumentata e Alternativa (ACC).

Nel capitolo 2 abbiamo visto che questi sistemi sono classificabili in tre categorie:

1. low tech (Tabelle)
2. middle tech (Comunicatori Elettronici e Meccanici)
3. high tech (PC adattati )

Al terzo gruppo appartengono i sistemi che permettono l'accesso alla comunicazione interpersonale in tempo reale, ma anche e soprattutto alla CMC "Classica": la generazione e l'inoltro di messaggi codificati nel formato ASCII. L'accesso a questi PC adattati avviene utilizzando strategie ed ausili alternativi alla tradizionale tastiera, quali la scansione, l'uso di tastiere virtuali e di un caschetto per digitare. Queste soluzioni seppur efficaci, raramente sono efficienti poiché consentono una velocità di trasmissione delle informazioni dall'utente alla macchina estremamente bassa.

### **8.2 Predizione del testo.**

I vari metodi usati per velocizzare l'interazione uomo-macchina sono stati esaminati nel capitolo 3. Ci siamo soffermati sui metodi di predizione: sistemi che suggeriscono all'utente le parole da scrivere permettendogli di premere un numero minore di tasti. Usualmente sono sistemi di tipo statistico, basati sulla diversa frequenza d'uso delle parole.

Nel 4° capitolo abbiamo analizzato come l'efficacia di questi metodi possa essere aumentata per mezzo di algoritmi di predizione sintattica che, servendosi di una conoscenza della grammatica e di un lexicon che suddivide le parole a seconda del loro tipo sintattico, cercano di identificare tra le parole probabili quelle grammaticalmente plausibili.

I metodi sintattici possono essere implementati servendosi di una Grammatica Libera dal Contesto. Si è visto però che questo sistema, adatto per l'analisi top-down di un testo, non è in grado di assicurare la correttezza sintattica di una frase del linguaggio naturale. Per ovviare a questo problema la CFG ed il relativo lexicon tipizzato devono essere aumentati.

Una seconda soluzione è rappresentata da una matrice di probabilità, in cui è codificata la probabilità che ad una parola di un certo tipo ne segua una di un altro tipo. Questo secondo metodo, più elastico, è più adatto alle applicazioni pratiche.

Un buon sistema di predizione non sfrutta solo informazioni di tipo sintattico, ma le associa a quelle di tipo statistico ed ortografico normalmente utilizzate dagli algoritmi di Word Prediction. Nel capitolo 5 si delinea l'algoritmo ottimale che un'applicazione di Word Prediction può implementare per applicare un sistema predittivo sintattico basato sul metodo della matrice dei pesi.

### **8.3 Il codice Mind Reader.**

Per verificare e sperimentare le precedenti ipotesi è stata sviluppata usando il linguaggio Delphi 3.0 un'applicazione per Windows 95 denominata Mind Reader. La scelta del linguaggio e dell'ambiente operativo è stata dettata dalla necessità di realizzare un'applicazione con un'interfaccia ergonomica ed in grado di funzionare sulla maggioranza dei computer in commercio. L'architettura del sistema è composta da tre parti che interagiscono ad ogni istante scambiandosi le necessarie informazioni:

- l'interfaccia
- il motore predittivo
- le basi di dati (lexicon + matrice dei pesi)

Oltre agli algoritmi di predizione ortografica, statistica e sintattica il programma implementa alcuni altri espedienti che permettono di aumentare il risparmio di tasti premuti.

Mind Reader è stato sottoposto ad una serie di test per valutare l'efficacia del sistema scelto e dell'algoritmo utilizzato, ed anche per controllare la reale efficienza dell'applicazione sul campo. Sono state implementate nel programma delle procedure in grado di automatizzare alcune valutazioni di efficienza.

## 8.4 Conclusioni.

I risultati ottenuti nei test preliminari indicano che Mind Reader ha un'efficacia comparabile con quella dei programmi commerciali disponibili (tabella 7.8).

Il programma si è rivelato di semplice utilizzo anche per utenti inesperti. E' dotato di numerose *features* che permettono una ottimizzazione rispetto alle esigenze dell'utente.

E' in progetto una sperimentazione dell'applicazione su un ampio numero di soggetti con diverse tipologie di disABILITÀ comunicative: è infatti possibile che questa applicazione possa rivelarsi utile per persone con disABILITÀ comunicative non solo di origine neuromotoria ma anche di tipo cognitivo (ad esempio afasie, vedi appendice A).

Concludiamo ricordando che diverse tecnologie oggi di uso comune, come la macchina da scrivere, la posta elettronica, il riconoscimento e la sintesi vocale, sono state inizialmente sviluppate come ausili comunicativi per persone con disABILITÀ. Anche la Word Prediction potrebbe diventare un giorno una funzione normalmente implementata in tutti i programmi di Word Processing per velocizzare la produzione dei testi da parte di persone non molto "abili" con la tastiera di un computer.



## **A - Classificazione delle disABILITÀ nella comunicazione verbale.**

Le disABILITÀ nella comunicazione verbale possono riguardare la creazione del messaggio, la sua produzione o la ricezione.

1. **DisABILITÀ sensoriali:** sono provocate da patologie degli organi sensoriali e comprendono ipovisione, daltonismo, cecità, sordità parziali e totali. Causano la diminuzione della capacità di ricevere la comunicazione verbale attraverso il canale sensoriale menomato.
2. **DisABILITÀ motorie:** sono provocate da patologie che alterano il controllo dei muscoli volontari, in particolare di quelli che presiedono alla produzione del parlato e dei testi scritti (manualità fine). Comprendono disartrie, anartrie e patologie neuromotorie (vedi paragrafo 1.4)
3. **DisABILITÀ intellettive:** sono provocate da disfunzioni congenite o acquisite del sistema nervoso centrale (SNC), e causano difficoltà nella generazione e nella decodifica dei messaggi verbali. Possono essere ulteriormente suddivise in:
  - a) Cognitive (insufficienza mentale);
  - b) Semantiche (afasia);
  - c) Mnemoniche;
  - d) Attentive;
4. **DisABILITÀ complesse:** comprendono i casi in cui più tipologie di disABILITÀ sono presenti contemporaneamente, si verificano in seguito a patologie del sistema nervoso centrale quali traumi cranici, ictus, paralisi cerebrali infantili, ecc.
5. **DisABILITÀ tecnologiche:** sono determinate da patologie che non impediscono la comunicazione verbale, ma l'accesso ai normali mezzi di trasmissione della comunicazione (telefono, computer, ecc.). Comprendono tutte le categorie precedenti ed altre tipologie di disABILITÀ in particolare di tipo motorio (paresi, amputazioni, ecc.).



## **B - Derivazione della matrice dei pesi usata da Mind Reader**

La matrice dei pesi *norm.pesi*, usata per i test dell'efficacia di Mind Reader (paragrafo 7.3.1), è stata ottenuta usando il metodo empirico delineato nel paragrafo 5.1.2.

La struttura standard di frase scelta è:

parte nominale + parte verbale + complementi

E' stata espansa nel modo seguente:

parte nominale

→ articolo + [aggettivo] + nome + [aggettivo]

→ articolo + infinito

parte verbale

→ verbo + avverbio

→ verbo essere + [avverbio] + aggettivo

→ verbo avere + [avverbio] + participio passato transitivo

→ verbo stare + gerundio

→ verbo servile + infinito

complementi

→ preposizione + parte nominale

→ preposizione + parte nominale

→ preposizione + infinito

→ preposizione + nome proprio

APPENDICE B – Derivazione della matrice dei pesi usata da Mind Reader

I tipi sintattici tra parentesi quadre sono considerati facoltativi. I participi passati sono considerati equivalenti agli aggettivi, le preposizioni articolate equivalenti agli articoli.

La matrice dei pesi (riportata in figura B.1) viene ricavata assegnando peso 3 o 2 alle coppie di tipi che compaiono nel modello (3 a quelle arbitrariamente considerate più frequenti, 2 alle altre), assegnando peso 1 a tutte le altre coppie e peso 0 alle coppie di parole dello stesso tipo (fatta eccezione per le coppie nome-nome e aggettivo-aggettivo). Le categorie sintattiche sono state ricavate dal riferimento 16

**Tabella B.1:** Matrice dei pesi “normal.pesi” per la lingua italiana.

	cl	st	inf	es	au	ha	Gr	v	av	pr	p2	cnj	Art	pa	np	n	ve	pr	ag	pp	p2	oth
Clitico	0	3	1	3	3	2	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1	1
v.stare	1	0	1	1	1	1	3	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Infinito	1	1	0	1	1	1	1	1	3	1	1	1	1	1	1	3	1	1	1	1	1	1
v.essere	1	1	1	0	1	1	1	1	3	1	1	1	1	1	1	1	1	1	3	3	3	1
Ausiliare	1	1	3	1	0	1	1	1	2	1	1	1	1	1	1	1	1	1	1	1	1	1
Avere	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	2	1	1	1	1	3	1
Gerundio	1	1	1	1	1	1	0	1	3	1	1	1	1	1	1	1	1	1	1	1	1	1
Verbo	1	1	1	1	2	1	1	0	3	1	1	1	1	1	1	2	1	1	1	1	1	1
Avverbio	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
Preposiz	1	1	1	1	1	1	1	1	1	0	1	1	1	1	3	3	1	1	2	1	1	1
Preposiz2	1	1	3	1	1	1	1	1	1	0	0	1	1	1	3	3	1	1	2	1	1	1
Conjuz	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Articolo	1	1	2	1	1	1	1	1	1	1	1	1	0	1	1	3	1	1	3	3	3	1
Prepartic	1	1	2	1	1	1	1	1	1	1	1	1	1	0	1	3	1	1	3	3	3	1
Nomepr	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Nome	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	2	1	1	3	3	3	1
v. venire	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	1
Pronome	1	2	1	2	2	2	1	2	1	1	1	1	1	1	1	1	1	0	1	1	1	1
Aggettivo	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	3	1	1	1	1	1	1
Partpass	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Partpass2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
Other	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

## **C - Software di Word Prediction in italiano per Windows.**

I software di Word Prediction sono ampiamente diffusi e utilizzati nei paesi anglosassoni: diverse applicazioni sono state sviluppate sia per scopi di ricerca che in campo commerciale [26]. Purtroppo questo tipo di programmi non sono conosciuti in Italia ed il numero di applicazioni sviluppate per la lingua italiana è veramente esiguo (vedi Tabella C.1).

Una prima possibilità per un utente italiano è l'utilizzo di programmi sviluppati per la lingua inglese e dotati di soli algoritmi alfabetici o statistici. Vi sono però due grossi inconvenienti: in primo luogo l'interfaccia e i manuali di queste applicazioni sono comunque in inglese, l'utente deve quindi conoscere questa lingua, e in secondo luogo questi programmi non sono dotati di un vocabolario iniziale. L'utente stesso deve crearlo.

Nell'ambito delle ricerche per questa tesi ho collaborato con una ditta canadese, l'Aurora System Inc. di Vancouver, alla realizzazione della versione italiana di Aurora 2.0, realizzando la traduzione dell'interfaccia e della manualistica e fornendo (grazie alla collaborazione del prof. Rodolfo Delmonte) un corpus di testi e una lista di parole in lingua italiana per la realizzazione del lexicon. La collaborazione con una ditta commerciale ha impedito la sperimentazione del programma ed una modifica del sorgente per adattarlo alla sintassi italiana: queste difficoltà mi hanno indotto alla realizzazione di un'applicazione direttamente progettata per l'italiano.

Altri due programmi per la WPD in italiano sono M.A.I.A (Multiple Access Input Assistant), realizzato dal Politecnico di Milano nell'ambito di un progetto coordinato dal prof. Somalvico, e Dedalus 2.0, un text editor commerciale per persone con disABILITÀ motoria che implementa algoritmi di predizione.

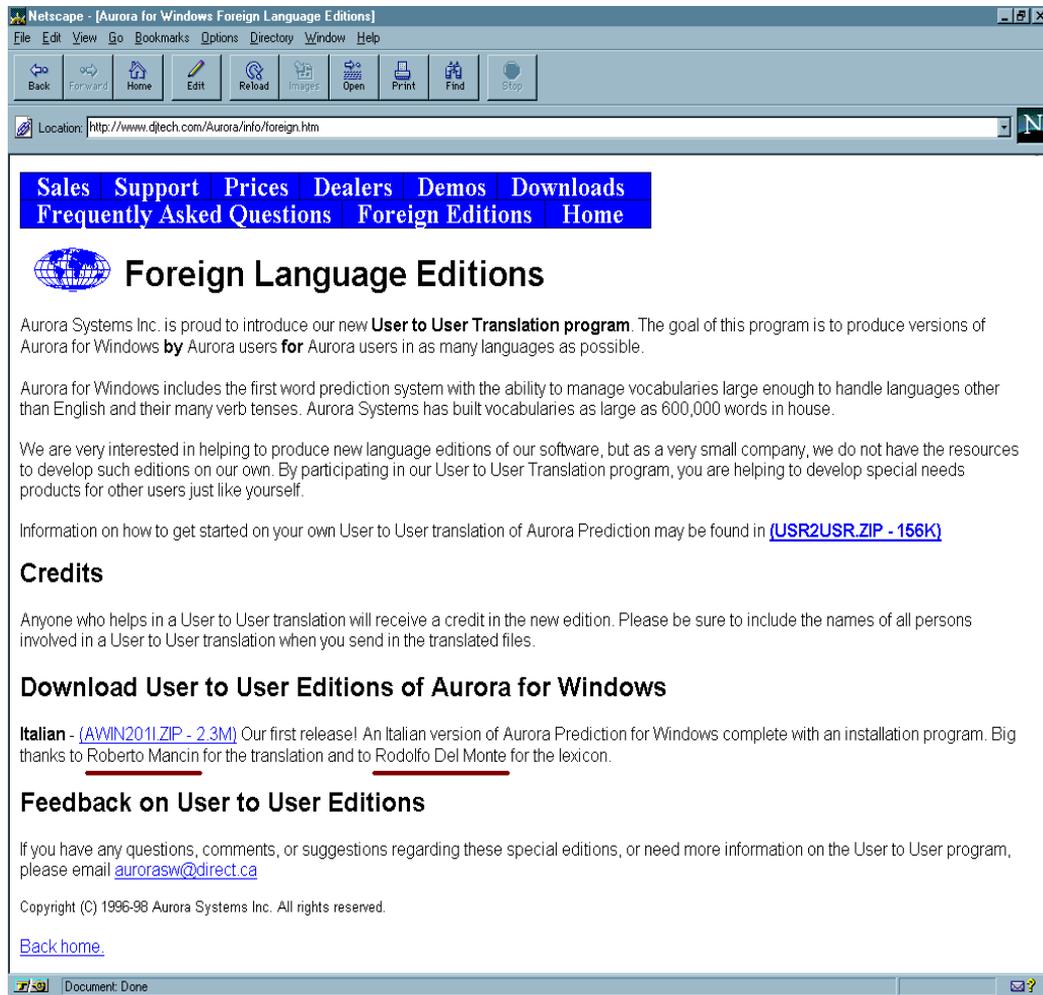
Le caratteristiche principali di questi tre programmi e di Mind Reader sono riportate in tabella C.1. La figura C.1 raffigura la pagina Web di Aurora 2.0 in cui si parla della versione italiana da me realizzata, le figure C.2 e C.3 mostrano l'interfaccia dei programmi Dedalus e Maia, infine in C.4 e C.5 sono raffigurate le due interfacce di Mind Reader.

Appendice C – Software di Word Prediction in italiano per Windows

**Tabella C.1:** Caratteristiche dei programmi di WPd in italiano

	<b>Aurora 2.0</b>	<b>Maia</b>	<b>Dedalus 2.0</b>	<b>MR 1.0</b>
piattaforma	win 3.1/95	win 3.1/95	win 3.1/95	win 95
word processor interno	no	no	si	si
aggiunta automatica di parole e frasi	si	si	no	si
scelta del dizionario	si	si	no	si
editing del dizionario	si	si	si	si
apprendimento automatico delle parole	si	si	no	si
apprendimento automatico delle freq..	si	no	no	si
espansione abbreviazioni	si	si	no	si
struttura dati del dizionario	?	lista ASCII	lista ASCII	tabella .dbf indicizzata
<b>finestra dei suggerimenti</b>				
scelta della dimensione dei caratteri	si	si	no	si
scelta del tipo di carattere (font)	si	no	no	no
<b>scelta delle parole</b>				
MaxSugg	1-9 + 6	4	5	1-9
scelta con tastiera	si	si	si	si
scelta con mouse	si	si	si	si
ordinamento alfabetico	si	si	no	si
ordinamento statistico	si	?	no	si
ordinamento sintattico	no	?	si	si
ordinamento sintattico e statistico	no	?	no	si
<b>opzioni</b>				
prima lettera della frase maiuscola	si	si	si	si
output vocale	si	si	no	no
input a scansione	no	si	si	no
spazio automatico	si	si	si	si
elimina ripetizioni	no	no	no	si
spazio dopo punteggiatura	si	si	si	si
tastiera "virtuale"	no	si	si	si
autore	?	Luca Clivio	Gianni Fadda	Roberto Mancin
realizzazione	Aurora system INC	UniMi	Cover Studio (FI)	UniPd UniVe

## Appendice C – Software di Word Prediction in italiano per Windows



**Figura C.1:** Pagina Web di Aurora 2.0 in cui si parla della versione italiana del programma (<http://www.djtech.com/Aurora/info/foreign.htm>).

Appendice C – Software di Word Prediction in italiano per Windows

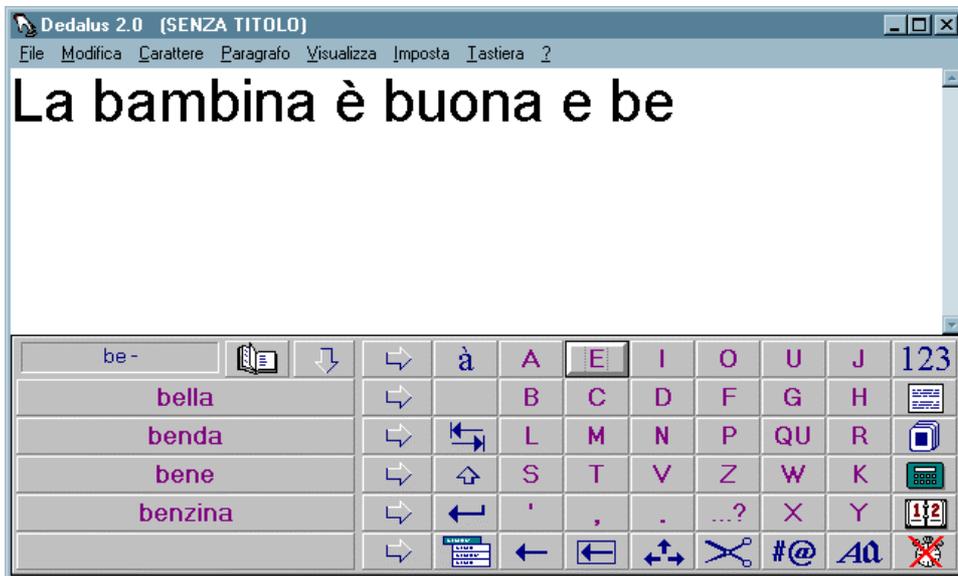


Figura C.2: Interfaccia di Dedaduls 2.0

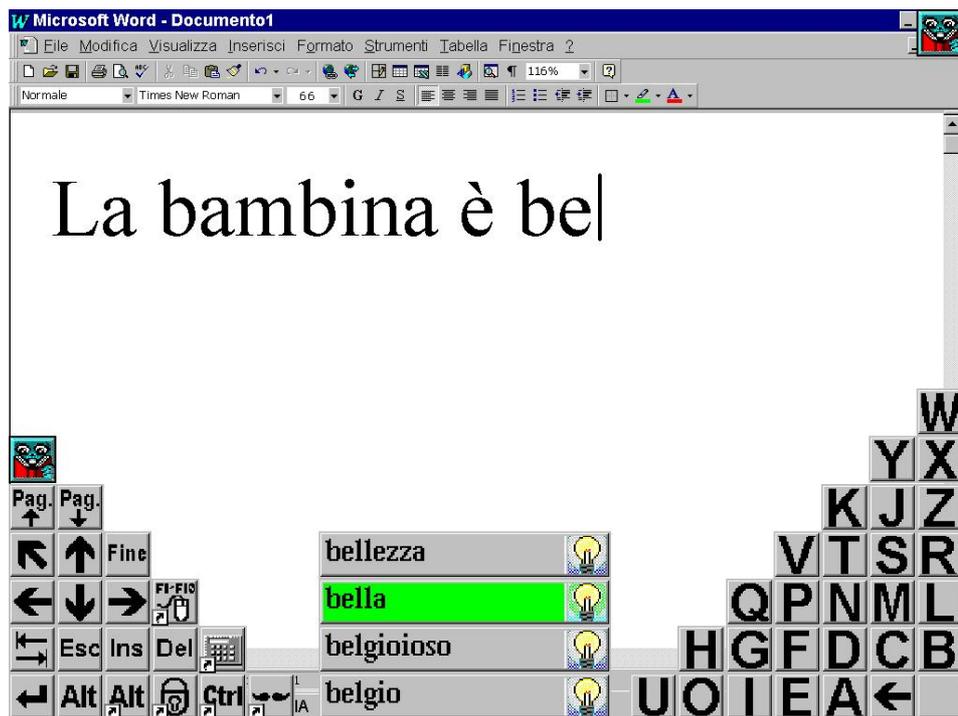


Figura C.3: Interfaccia di MAIA

Appendice C – Software di Word Prediction in italiano per Windows

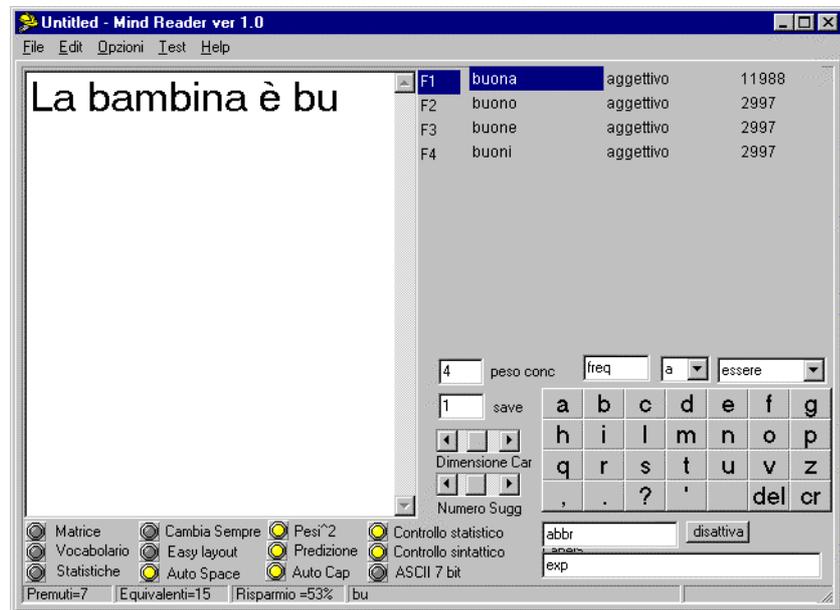


Figura C.4: Interfaccia completa di Mind Reader



Figura C.5: Interfaccia semplice di Mind Reader



## **D - Glossario**

### **Applicazione**

Programma sviluppato per assolvere particolari funzioni (ad es. a livello gestionale, scientifico, istruzionale, ecc.).

### **Ausilio**

Qualsiasi prodotto, strumento, attrezzatura o sistema tecnologico di produzione specializzata o di comune commercio, utilizzato dalla persona disABILE per prevenire, compensare, alleviare o eliminare una menomazione, disABILITÀ, handicap. Non si tratta di attrezzatura con finalità clinica.

### **Ausilio informatico**

Ambiente informatico, o sistema hardware o software per l'interazione fra persona e l'ambiente informatico, utilizzato dalla persona disABILE per prevenire, compensare, alleviare o eliminare una menomazione, disABILITÀ, handicap.

### **Bliss Symbol**

E' un sistema costituito da simboli scritti che, indicati di volta in volta dall'utente, permettono di comunicare all'interlocutore il suo pensiero. E' basato sul significato e non sulla fonetica: ogni simbolo, che costituisce un "ideogramma", esprime un concetto o parte di esso.

Combinando i simboli si possono esprimere concetti di crescente complessità.

### **Codifica**

Metodo che consente di esprimere un elevato numero di significati a partire dalla combinazione di un numero limitato di elementi: ad es. diversi emulatori di tastiera per PC prevedono l'uso del codice Morse (combinazione di tre elementi: punto, linea, pausa) per esprimere tutti i caratteri presenti sulla tastiera standard e per richiamare macro-istruzioni (frasi già pronte, sequenze di comandi, ecc.).

**Comunicatore**

Dispositivo in grado di trasformare un codice comunicativo particolare in un messaggio intellegibile per l'ambiente circostante. Nel linguaggio corrente individua un dispositivo, spesso non legato al PC, per la comunicazione in presenza dell'interlocutore. Normalmente i comunicatori presentano facilitazioni per l'input (scansione, codifica) e si suddividono in simbolici e alfabetici.

**Configurazione hardware**

Definisce le caratteristiche del sistema di elaborazione e delle sue periferiche. La configurazione hardware non è riferita a un particolare modello di computer; lo stesso tipo di elaboratore infatti può avere configurazioni diverse a seconda della quantità di memoria centrale, del numero dei disk drive, del tipo di scheda grafica, del tipo di monitor (monocromatico o a colori) che utilizza, ecc.

**DisABILITÀ**

Limitazione più o meno grave della capacità di assolvere compiti e soddisfare bisogni in modo adeguato a causa di menomazioni o danni, singoli o multipli.

**Emulatore**

Dispositivo che consente di svolgere con modalità alternative le funzioni svolte da un dispositivo standard. ES.: emulatore di tastiera (che fornisce sistemi di input al PC alternativi alla tastiera), emulatore di mouse, ecc.

**Feedback**

Informazione di ritorno prodotta dal dispositivo verso l'utilizzatore durante l'azionamento. Esempi possono essere: il rinforzo sonoro associato ad una scelta all'interno di un software didattico, il click associato alla pressione di un tasto, ecc.

**Handicap**

Condizione di svantaggio esistenziale conseguente a una disABILITÀ che impedisce di ricoprire in modo adeguato il ruolo che la società si attende dal soggetto.

**Interfaccia**

Modalità che si interpone come mediazione per consentire la relazione tra due entità. Nel campo degli ausili e delle apparecchiature tecnologiche l'interfaccia può essere costituita da dispositivi hardware e/o software nella fase di input (fra persona-strumento: ad es. un sensore speciale che consenta da una persona disABILE di azionare oggetti altrimenti non utilizzabili), nella fase di elaborazione del segnale (interfaccia fra strumenti) o nella fase di output (interfaccia strumento-ambiente esterno).

**Joystick**

Dispositivo di input composto da una leva che è possibile spostare in tutte le direzioni. In uscita può ottenere la semplice chiusura di uno o più contatti elettrici (Joystick digitale) oppure dei valori che indicano la posizione della leva (Joystick analogico).

**Menomazione**

Il danno oggettivo subito dalla persona: qualsiasi perdita o anomalia a carico di strutture o funzioni psicologiche, fisiologiche o anatomiche.

**Mouse**

E' un dispositivo di input dotato di uno o più pulsanti. Al suo movimento su una superficie piana corrisponde il movimento di un puntatore sullo schermo del computer. E' utilizzato per selezionare parti dello schermo corrispondenti a specifiche funzionalità (ad esempio le opzioni di un menu) o per modificare le informazioni contenute sullo stesso schermo (ad esempio un disegno).

**Multimediali**

Programmi che consentono di utilizzare più canali sensoriali contemporaneamente per trasmettere i "messaggi". I migliori presuppongono computer sofisticati in grado, fra l'altro, di riprodurre il suono con qualità hi-fi e di visionare brani animati e filmati.

**Periferica**

Qualsiasi dispositivo che può essere collegato ad un calcolatore come unità di output o input. (Es.: Stampanti, monitor, Tastiere, dispositivi speciali per disABILI, ecc..).

**Posta Elettronica**

Scambio asincrono di messaggi testuali tra utenti di una rete telematica.

**Puntatore**

Elemento direzionabile sullo schermo, normalmente rappresentato da una freccetta (v. anche sistema di puntamento)

**Riconoscitore di voce**

Sistema hardware e software in grado di riconoscere la voce umana dopo una fase di addestramento da parte dell'utente. Consente di dare comandi al computer tramite la voce senza usare la tastiera.

**Scansione**

Particolare sistema per la selezione di oggetti che propone gli elementi da scegliere uno alla volta in sequenza. La scelta avviene tramite codifica SI/NO. La scansione degli oggetti può avvenire secondo varie modalità (automatica, semiautomatica, manuale, ecc..).

**Selezione diretta**

Individua la possibilità di indicare direttamente l'elemento desiderato fra molti. Per quanto riguarda ad esempio la comunicazione scritta, la selezione diretta fa riferimento all'indicazione con il dito o con lo sguardo su un alfabetiere, l'uso della tastiera e/o del mouse sul computer, ecc.

**Sensore**

Dispositivo che trasforma una grandezza fisica (es. pressione, spostamento, suono) in una elettrica utilizzabile per comandare dispositivi (giocattoli, computer, ecc.).

**Sintesi Vocale**

Tecnica che consente di sintetizzare la voce umana tramite un calcolatore e di usarla come output in sostituzione dei sistemi tradizionali quali monitor, stampanti, ecc.

**Sintetizzatore vocale**

Dispositivo in grado di riprodurre con voce artificiale qualsiasi testo memorizzato in un Personal Computer: può essere basato su una scheda da inserire all'interno del computer o su apparecchiature esterne collegabili attraverso la porta seriale e la parallela.

**Sistema di puntamento**

Dispositivo di input basato sul controllo di un puntatore che si muove liberamente sullo schermo. Viene spesso associato a uno o più tasti per consentire la selezione degli oggetti puntati. Il sistema di puntamento più comune è il mouse.

**Switch**

Commutatore elettrico Acceso/Spento (v. Sensore).

**Tasti funzione**

Tasti che non producono fisicamente sul video lettere, numeri o simboli ma che hanno, o possono avere, funzioni diverse a seconda del programma che si sta eseguendo. In genere sono contrassegnati dalla lettera F seguita da un numero (F1, F2, ...).

**Tastiera espansa**

Tastiera di grandi dimensioni per computer o per macchina da scrivere, con i tasti ingranditi, ben separati tra loro e spesso caratterizzati da una diversa sensibilità rispetto alla tastiera standard.

**Tastiera ridotta**

Tastiera speciale caratterizzata da dimensioni ridotte e tasti piccoli, ravvicinati e molto sensibili. Viene utilizzata nei casi in cui si hanno movimenti deboli e limitati ma precisi.

**Telematica**

Telecomunicazione per la trasmissione a distanza di dati: le reti telematiche rendono possibile la comunicazione fra computer.

**VOCA - Vocal Output Communication Aids**

Ausilio per la comunicazione in grado di generare messaggi sonori.

## BIBLIOGRAFIA.

## Bibliografia.

- [1] C. BITELLI, V. VIGLIETTI, *L'ausilio per cominciare*, Tecnologia e integrazione, software didattico, ausili e telelavoro, Graffito snc, Bologna, 1996.
- [2] C. CUNNINGHAM, N. COOMBS, *Information Access and Adaptive Technology*, Oryx press, Phoenix, 1997.
- [3] J.S. SHANDU, T. WOOD, *Demography and market sector analysis of thirteen European countries*, Newcastle Polytechnic, Newcastle, 1990.
- [4] P. MORASSO, P. GIANNONI, *Ausili di comunicazione per disABILI motori*, Protesi e ausili per la comunicazione, Pàtron editore, Bologna, 1995, pp 147-160.
- [5] S. HAWKING, *My Experince with ALS*,  
<http://www.damtp.cam.ac.uk/user/hawking/diasbilty.html>.
- [6] D. R. BEUKELMAN, P. MIRENDA, *Augmentative and Alternative Communication*, Paul Brookes Publishing, Baltimore, 1991.
- [7] C. ZANGARI, L. LLOYD, B. VICKER., *Augmentative and alternative communication: A historical perspective*, *Augmentative and alternative communication*, 1994, 10(1), pp.105-106.
- [8] M. DE GASPERI, *Sordità ed accessibilità delle risorse scientifiche e matematiche*, Atti del 5° Convegno Nazionale Informatica e DisABILITÀ (IDD'97), Bologna, 1997, pp.155-157.
- [9] *Introduction to Augmentative and Alternative Communication (AAC)*  
<http://call-centre.cogsci.ed.ac.uk/callresearch/AAC/AACintro>.
- [10] R. MANCIN, F. SARTORETTO, *DisABILITÀ e multimedialità: come migliorare l'interazione*, Atti del 9° Congresso Nazionale di Informatica Medica (AIIM'97), Venezia, 1996, pp. 181-184
- [11] M. E. WOOD, *Syntatic Pre-Processing in Single-Word Prediction for Disabled People*, Tesi di Dottorato, University of Bristol, Bristol, 1996.
- [12] N. CHOMSKY, *Syntactic Stuctures*, Mouton, The Hague, 1957.
- [13] J.E. HOPCROFT, J.D. ULLMAN, *Introduction to Automata Theory Languages and Computation*, Addison-Wesley, Reding, 1979.

## BIBLIOGRAFIA.

- [14] C. MORRIS, L. BOOTH, I. RICKETTS, N. ALM, A. NEWELL, *Evaluation of a syntax-driven word predictor for children with language impairments*, Proceeding of the Annual Conference of RESNA '93, Washington, 1993, pp. 423-425.
- [15] P. LIGHT, C. COLBOUM, *Verso il cielo: La Comunicazione Mediata da Computer a supporto dei corsi universitari tradizionali*, tecnologie didattiche, 12, pp. 50-55
- [16] R. DELMONTE, *Lessico e Grammatica. Teorie Linguistiche e applicazioni lessicografiche*, Bulzoni, Roma, 1997.
- [17] J. KLUND, M. NOVAK, *If Word Prediction Can Help, Which Program Do You Choose?*, [http://www.trace.wisc.edu/docs/word\\_pred\\_prog/ctg\\_wp.htm](http://www.trace.wisc.edu/docs/word_pred_prog/ctg_wp.htm).
- [18] G. C. VANDERHEIDEN, *Application Software Design Guidelines*, 1994, [http://www.trace.wisc.edu/docs/software\\_guidelines/toc.htm](http://www.trace.wisc.edu/docs/software_guidelines/toc.htm).
- [19] TIDE USER CONSORTIUM, *USERfit: a practical handbook on user-centred desing for Assistive Tecnology*, ECSC-EC-EAEC, Brussels-Luxemburg, 1996.
- [20] J. PREECE, *A Guide to Usability*, Open University Press, Walton Hall, 1990.
- [21] C. COLOMBO, A. DEL BIMBO, *Interacting through eyes*, <http://bising.ing.unibs.it/~columbus/columbications.html>
- [22] D.J. HIGGINBOTHAM, *Evaluation of keystroke saving across five assistive communication technologies*; *Augmentative and Alternative Communication*, 1992, 8, pp 258-272.
- [23] D.K. ANSON, *The effect of Word Prediction on Typing Speed*; *Am. J of Occunpation Therapy*, 1993, 47(11), pp 1039-1044.
- [24] M. CANTU , *Mastering Delphi 3*, SYBEX, Alamenda, 1997.
- [25] R. MANCIN, *VOCA: Ausilio portatile per la comunicazione con uscita in voce sintetica*, Atti del 4° Convegno Nazionale Informatica e DisABILITÀ (IDD'95), Bologna, 1995, pp 395-400.  
[http://www.comune.rom.it/COMUNE/sperimentali/cam/gr/idd95/395\\_g.html](http://www.comune.rom.it/COMUNE/sperimentali/cam/gr/idd95/395_g.html)
- [26] *Glossary of Adaptive Tecnologies: Word Prediction*, <http://www.utoronto.ca/atrc/tech/wordpredsr.html>.
- [27] D. METZING, *Augmented transition networks used in a procedural dialogue model*, Proceedings of the interantional Conference on Computational Linguistics (COLING 80), Association for Computational Linguistics, Tokyo, 1980, pp. 487-491.

## BIBLIOGRAFIA.

- [28] M. SOMALVICO, L. CLIVIO, G. VENTURIN. *MAIA: Interfaccia per l'utilizzo di un personal computer con dispositivi di input alternativi in ambiente grafico*, Milano 1995  
<http://www.elet.polimi.it/section/compeng/air/ihd/maia/maia.htm>
- [29] N. CHOMSKY, *Aspects of the Teory of Syntax*, MIT Press, Cambrige (Mass.), 1965



## Indice analitico.

<b>A</b>	
AAC <i>Vedi</i> Augmentative and Alternative Communication	
Abbreviation expansion.....	22
<i>American Standard Code for Information Interchange</i> ..... <i>Vedi</i> ASCII	
ASCII.....	9; 18; 19; 20; 49; 57; 83; 92
Augmentative and Alternative Communication.....	11; 12
<b>B</b>	
Bliss.....	12; 17; 97
<b>C</b>	
CFG..... <i>Vedi</i> Context Free Grammar	
CMC <i>Vedi</i> Computer Mediated Communication	
Communication System	
Low Tech.....	12
Medium Tech.....	14
High Tech.....	15
Completamento..... <i>Vedi</i> Word completion	
Computer Mediated Communication.....	15
Comunicazione	
asincrona.....	17; 20
interpersonale.....	12; 78; 83
sincrona.....	20
testuale.....	4; 15; 16; 18; 24
<b>ConcAtt</b> .....	40; 42; 60; 66
Concordanza.....	37
Context Free Grammar.....	31; 34
<b>D</b>	
DisABILITÀ	
intellettive.....	5
motorie.....	4
nella comunicazione verbale.....	5
uditive.....	5
visive.....	4
Dizionario	
dinamico.....	26
multiplo.....	26
statico.....	25
<b>F</b>	
Feedback.....	13; 79
<b>G</b>	
Grammatica Libera dal Contesto, <i>Vedi</i> Context Free Grammar	
<i>Graphic User Interface</i> ..... <i>Vedi</i> GUI	
GUI.....	4; 18
<b>K</b>	
<i>Keystroke_saving</i> .....	23

## INDICE ANALITICO

<b>L</b>	
Lexicon .....	32, 40
LIS .....	12
<b>M</b>	
Matrice dei pesi.....	36; 41
<b>MaxSugg</b> 42; 44; 60; 66; 69; 70; 71; 72; 73; 74; 75; 77; 80; 92	
Multi Word Prediction .....	
<i>Vedi</i> Predizione frase	
<b>N</b>	
Noam Chomsky.....	30
<b>P</b>	
<b>ParCur</b> .....	41; 44; 45; 60; 66
Patologie neuromotorie	
Distrofia muscolare .....	7
Ictus cerebrale .....	8
Lesione midollare .....	8
Paralisi cerebrali infantili .....	6
Sclerosi laterale amiotrofica (ALS) .....	7
Sclerosi multipla (MS) .....	7
Trauma cranico (TBI) .....	8
Phrase completion .....	22
Predizione	
alfabetica .....	25
carattere.....	27
<b>E</b>	
Espansione di acronimi-abbreviazioni	
.....	27
frase .....	28
parola .....	28
semantica .....	26
sintattica debole .....	26; 35
sintattica forte .....	26; 29
statistica .....	25
<b>R</b>	
riconoscimento vocale .....	5; 19; 26; 40; 79
<b>S</b>	
Scansione ....	14; 19; 24; 27; 64; 83; 98; 100
Semantic encoding .....	21
Semantica .....	30
Sintassi .....	30
Stephen Hawking .....	7
Struttura standard di una frase .....	41, 89
Switch .....	19; 24
<b>T</b>	
Tabella alfabetica .....	13
<b>TipoPrec</b> .....	42; 54; 60; 66
<b>W</b>	
WindMill .....	39
Word completion .....	22
Word prediction .....	22