Luigi Rizzi (rizzil@unisi.it) - Cristiano Chesi (chesi@media.unisi.it) 26 Febbraio 2003

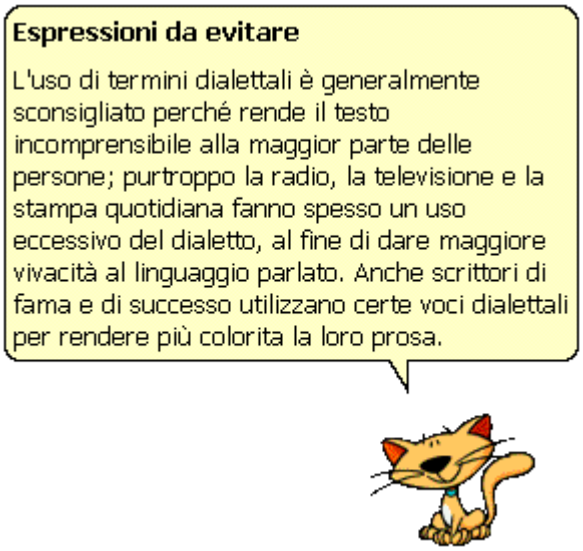
INTRODUZIONE AL NLP: APPLICAZIONI E PROBLEMI

Alcuni esempi di Natural Language Processing

Eliza (Weizenbaum, **anno 1966**):
Utente: il mio ragazzo dice che sono sempre depressa
Eliza: sono spiacente di sapere che sei depressa

HAL 9000 (Kubrick & Clarke, 2001 Odissea nello spazio; **anno 1968**):
David: Apri la saracinesca esterna, Hal.
Hal: Mi dispiace David, purtroppo non posso farlo.

Correttore Grammaticale di Microsoft Word eXPerience (Expert System, **anno 2003**)
Utente: "voglio veduto Mario al posto mio" (inteso "voglio vedere Mario al posto mio")
Arturo: questa forma dialettale deve essere sostituita con l'equivalente in italiano. Sostituire "voglio veduto" con "devo essere veduto" oppure "vado veduto"
Utente: [preme il tasto "spiega…" che fornisce una spiegazione dell'errore]
Arturo:



(1) cosa avrebbe dovuto saper fare HAL 9000:
   a. **speech recognition / synthesis** – analisi/produzione del segnale acustico, identificazione delle formanti, sillabazione, suddivisione in parole, identificazione contorni prosodici
   b. **natural language understanding / generation** – trasformazione dell'informazione linguistica recuperata in (1) in un formato simbolico rilevante per le elaborazioni successive; tradizionalmente si identificano le seguenti fasi di analisi:
      i. **morfologia** – scomposizione delle parole in unità minime di significato (dogs = dog + s)
      ii. **sintassi** – definizione delle relazioni strutturali tra parole
      iii. **semantica** – attribuzione del significato delle espressioni
      iv. **pragmatica** – attribuzione di intenti in base agli usi/convenzioni linguistiche
      v. **discorso** – recupero di relazioni tra unità linguistiche più ampie della singola frase
   c. **information extraction** – identificazione delle porzioni di testo/conoscenza in cui risiede l'informazione rilevante
   d. **inferenza** – trarre le adeguate conseguenze dalle informazioni recuperate

(2) cosa si cerca di fare attualmente (e quanto bene lo si fa):

   a. **sillabazione** (soddisfacente)
   es. casa > ca-sa

b. **correzione ortografica** (soddisfacente)
es. cawa > casa

c. **correzione grammaticale** (povera)
es. lo casa > la casa

d. **correzione stilistica** (pessima)
es. mi trovai per una selva oscura > ero in un bosco buio

e. **riconoscimento del parlato** (soddisfacente)
es. /kasa/ > casa

f. **filtraggio/recupero di informazioni** (dipende dal contesto. In contesti ristretti in genere è soddisfacente)
es. "nel 2002 la Enron è fallita" > società: Enron; stato: fallimento; periodo: 2002

g. **rispondere a domande** (dipende dal contesto. In contesti ristretti in genere è soddisfacente)
es. dove si trova la penna? > sul tavolo

h. **ricerca "intelligente" su web** (soddisfacente, ma usa anche euristiche extralinguistiche)
es. capire se il tipo di ricerca è espansiva (cioè volta alla raccolta di molte informazioni) o puntuale (si cerca cioè solo la risposta ad una domanda)

i. **riassunto automatico e classificazione di un testo** (insoddisfacente)
es. "Avevo appena finito di tagliare il lesso (parecchio filaccioso, a dire la verità), quando nel rimettermi a sedere osservai, con una disposizione di spirito poco in carattere col mio abito, che chiunque si fosse preso la briga di eliminare il colonnello Protheroe avrebbe reso un gran servigio all'umanità".
La morte nel Villaggio, A. Christie
> il protagonista, finito di tagliare il lesso pensò che pochi si sarebbero dispiaciuti della morte del colonnello P.

j. **pseudo-comprensione** (dipende dal contesto. In contesti ristretti in genere è soddisfacente)
potresti chiudere questa finestra? > [chiusura della finestra di Word in questione]

k. **generazione del linguaggio naturale** (dipende dal contesto. In contesti ristretti in genere è soddisfacente)
[contesto precedente] > ho chiuso la finestra di Word a cui ti riferivi

l. **traduzione automatica** (soddisfacenti traduzioni parola per parola; grosse difficoltà di disambiguazione)
[contesto precedente] > I closed the Word window you pointed out.

Aspetti linguistici che vorremmo descrivere computazionalmente

(3) Due livelli di analisi: definire la **natura del problema** (**competence**) e capire come **risolverlo** (**processing**).

(4) **data-structure**: di che tipo di struttura dati ha bisogno la conoscenza linguistica?
a. una parola in italiano può iniziare per *ma...(mare)* ma non per *mr...*
b. la *e* di cas*e* ha un valore diverso da quella di mar*e*
c. "le case sono sulla collina" Vs. *"case le collina sono sulla"
d. il gatto morde il cane > sogg: gatto(agente); verbo: morde(azione); ogg: cane(oggetto)
e. ?il tostapane morde il gatto
f. l'espressione "le case" si riferisce ad un gruppo di case evidente dal contesto (Vs. "delle case")
g. ...

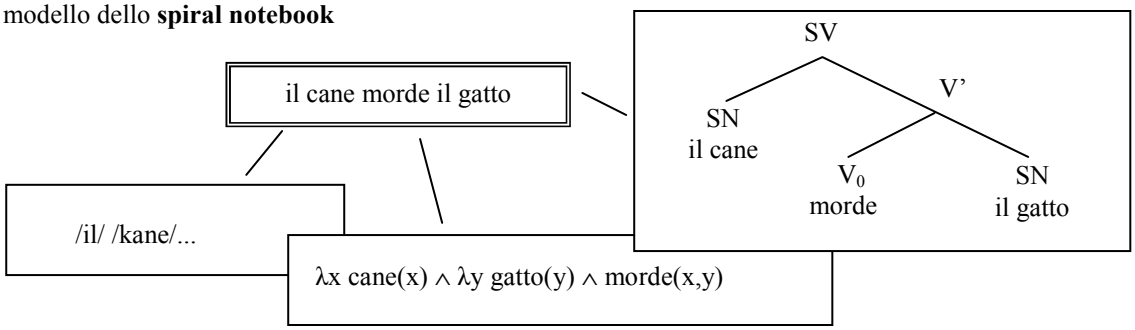
ad ogni livello si devono specificare delle primitive elementari:
fonemi tratti segmentali e soprasegmentali
morfemi identificazione delle regole combinatorie
parole gruppi di morfemi significativi
sintagmi gruppi tipizzati di parole che esprimono relazioni
elementi tematici quali paziente, agente...
elementi discorsivi convenzioni discorsive, relazioni pragmatiche pertinenti…

(5) …e precise specifiche di combinazione; definiremo cioè che tipo di **processing** è necessario per usare la conoscenza codificata dalla struttura dei dati.
ad esempio in fonologia definiremo delle restrizioni fonotattiche che impediranno la combinazione di certe concatenazioni di tratti fonemici o la riduzione di determinate sequenze in altre, in morfologia definiremo le regole di combinazione morfofonemiche che permetteranno, ad esempio in italiano, di flettere mangi*are* in mangi*ato* e sap*ere* in sap*uto*.

Un esempio storico (probabilmente il primo): Panini (400-600AC) descrive il sanscrito usando una serie di "regole di produzione" sotto forma di aforismi (sutra): partendo da circa 1700 elementi base suddivisi in classi (nomi, verbi ecc.) e indicando le regole di combinazione (circa 4000), si riusciva (almeno teoricamente) a derivare ogni forma accettabile in sanscrito.

(6) **processing** è diverso da **performance**; quest'ultima mostra come il processamento linguistico possa venire turbato da fattori extralinguistici (es. limitazione della memoria a breve termine).

(7) il modello dello **spiral notebook**



ogni livello è definito in termini di una relazione di mappatura con gli altri livelli.

(8) La **complessità del problema** deriva dal fatto che la mappatura non è sempre univoca:
a. **ambiguità lessicale** (la vecchia legge la regola)
b. **ambiguità sintattica** (ho visto il ragazzo nel parco con il cannocchiale)
c. **ambiguità semantica** (la pesca non è stata fruttuosa)

morale: un problema è più difficile se contemporaneamente devo valutare più possibilità, tutte ugualmente plausibili. Scelte multiple tra cui non ho euristiche di scelta portano al **non-determinismo**.

(9) **Parsing**
mettere in relazione un input con un'appropriata descrizione strutturale

Prime semplici applicazioni: la correzione ortografica

(10) **correzione ortografica** è diversa dal **controllo ortografico**: mentre il controllo può limitarsi semplicemente ad accettare/rifiutare una stringa di testo, la correzione deve proporre una forma corretta in alternativa.
Esempio di approccio ingegneristico:
definizione precisa del problema (classificazione degli errori) > ricerca di soluzioni adeguate ed efficienti

(11) all'identificazione degli errori tipici segue solitamente una **categorizzazione** su quattro livelli: lessicale, sintattico, semantico e pragmatico. Va ricordato che ogni errore può essere riconosciuto come tale sia perché è un vero errore (**malformatezza assoluta**), sia perché il sistema non è in grado di trattare, per la limitatezza delle risorse linguistiche utilizzate, la forma che in realtà sarebbe corretta (**malformatezza relativa**)

a. malformatezze lessicali

Relative
- parole non presenti nel lessico del sistema

Assolute
- tipografiche (omissioni, sostituzioni, inserzioni involontarie non proprie della parola)
- cognitive (errata credenza sull'ortografia della parola)
- fonetiche (errata credenza sull'ortografia della parola in base alla sua pronuncia)

b. malformatezze sintattiche

Relative
- inadeguatezza della teoria sintattica implementata (poche regole > ipergeneralizzazione; troppe regole > inconsistenza, esclusione di strutture in realtà corrette)
- forme colloquiali o dialettali (espressioni idiomatiche, indicativo al posto del congiuntivo…)
- pronomi di ripresa (pro-sintagmi ripetuti impropriamente)

Assolute
- pronome sbagliato (es. me sono andato)
- mancanza di accordo tra:
    - soggetto - verbo (es. Loro sono andato...)
    - modi - tempi (es. Voglio vado; vorrei andato)
    - determinanti - nomi (es. Lo casa)
    - aggettivi - nomi (es. Il mare verdi)
- omissioni di argomenti obbligatori (es. ho messo sul tavolo _)

c. malformatezze semantiche

Relative
- relazione non presente (relazioni tra gli oggetti non disponibili nella base di conoscenze del sistema)
- violazione delle restrizioni di selezione (uso di espressioni che violano le restrizioni della base di conoscenze del sistema)
- sinonimia (mancanza del collegamento semantico tra due sinonimi)
- polisemia (significati alternativi non presi in considerazione dal lessico di macchina)

Assolute
- violazione delle restrizioni di selezione (es. il telescopio nuotò a bordo)
- logica spaziale (es. vieni là)
- logica temporale (es. domani sono andato a ballare)

d. usi figurativi nella frase

- metafora (es. "con un filo di voce" per "con voce flebile")
- metonimia (es. "quel ferro vecchio va rottamato" per "quella macchina")
- sineddoche (es. "il mondo ci è nemico" per "si percepisce una certa ostilità")
- antonomasia (es. "il divino poeta" per "Dante")
- perifrasi (es. "quel coso per asciugare i capelli" per "asciugacapelli")
- eufemismo (es. "passare a miglior vita" per "morire")
- litote (es. "non è certo un'aquila" per "non è molto intelligente")
- iperbole (es. "l'ho detto mille volte" per "l'ho già detto molte volte")
- idioma (es. "il dado è tratto" per "ormai la decisione è stata presa")

(12) Le varie tecniche che permettono di gestire le malformatezze si basano principalmente su un sistema di **pattern matching** con il lessico di cui dispone il sistema e su una serie di **euristiche** per decidere le correzioni possibili alle forme errate tra cui:

a. Distanza minima
Il sistema inventato da Damerau (Damerau 64) e perfezionato da Wagner (Wagner 74) tratta l'errore come una forma che si differenzia da quella corretta per un numero minimo di operazioni di **inserimento**, **cancellazione**, **sostituzione** e **scambio** di caratteri.
Il metodo consiste nel calcolare attraverso una funzione diversa da sistema a sistema, la minima distanza di correzione tra le stringhe ortograficamente scorrette e le parole presenti nel vocabolario. Se questa distanza è considerata accettabile il vocabolo è considerato come possibile correzione della forma non standard.
Il grave difetto di questo approccio è l'inefficienza: l'elaborazione richiede un numero *n* di confronti, con *n* uguale al numero delle parole del vocabolario. Se l'euristica per calcolare la distanza è "buona" (difficile a priori stabilire un criterio di bontà) questa tecnica consente comunque di correggere un elevato numero di errori singoli (stima intorno al 95%).

b. Chiave di somiglianza
Questa tecnica associa ad ogni stringa una chiave costruita in modo che tutte le parole scritte o pronunciate in un modo simile abbiano una chiave uguale o molto somigliante.

Confrontando, non le parole, ma solo le chiavi si ottengono le candidate alla correzione della parola scorretta.
Il sistema è stato ideato da Odell e Russel (18) per la correzione di errori fonetici e successivamente è stato migliorato da Davidson (62).
Nella prima versione la chiave era formata dalla prima lettera della parola e da una sequenza di numeri associati secondo certe regole e statistiche di frequenza (es. vocali = 0; b, f, p, v = 1; altre consonanti = 2).
Gli zero e i numeri ripetuti venivano poi eliminati (es. casa = 2020 > c2; csa = 220 > c2).

Pollock e Zamorra (84) migliorano ulteriormente il metodo attribuendo due tipi di chiavi ad ogni parola del vocabolario:
una **skeleton key** formata dalla prima lettera della parola, dalle consonanti nell'ordine in cui si presentano nella parola senza ripetizioni e dalle vocali, sempre nell'ordine e sempre senza ripetizioni (es. gambero = gmbraeo);
una **omission key** formata dalle consonanti, senza ripetizione in un ordine di frequenza (determinato staticamente) e poi dalle vocali, senza ripetizioni, nell'ordine in cui si presentano nella parola.
Queste chiavi sono giustificate da una serie di osservazioni sulla distribuzione degli errori:

i. l'ordine delle vocali è spesso mantenuto invariato
ii. raramente viene sbagliata la prima lettera della parola, ma statisticamente gli errori si situano verso la fine del lessema scritto

Il sistema che vedeva implementato questo metodo (SPEEDCOP) consultava un dizionario comune di termini errati, applicava in ordine la chiave scheletro e la chiave omissione ed infine controllava le parole concatenate tra loro secondo una determinata funzione di somiglianza.
Questo approccio gestiva il 94% degli errori singoli e tra il 74% e l'88% degli errori complessivi presenti nel testo.

c. Regole

La tecnica basata su regole utilizza algoritmi ed euristiche per rappresentare la conoscenza necessaria per determinare quali sono le regole che il termine sbagliato ha violato e le correzioni necessarie per correggerlo (es. restrizioni fonotattiche + informazioni sull'ordine delle lettere sulla tastiera).
Una volta applicate tutte le regole a disposizione, i risultati vengono presentati all'utente secondo una stima di probabilità.
Il sistema realizzato da Yannakoudakis e Fawthrop (83) permette una precisione intorno al 76% di errori rilevati. Means (Means 88) affina la tecnica inserendo nel suo correttore oltre alle regole della morfologia inglese altre regole di abbreviazione e flessione non standard migliorando in parte i risultati del primo prototipo.

d. Analisi con n-grammi

L'idea da cui nasce questa tecnica è che una parola possa essere suddivisa in un insieme di piccole sottostringhe, gli n-grammi appunto, che si sovrappongano all'interno del lessema.
Ogni n-gramma porta con sé alcune informazioni sull'identità della parola.
Il vocabolario, seguendo questo approccio (Kohonen 80; DeHer 82; Angell et al. 83; DeSmedt e VanBerkel 88), deve essere strutturato come una tabella di n-grammi; ogni n-gramma rappresenta un indice che rinvia ad un determinato termine nel vocabolario di macchina. L'insieme dei rinvii determina il campo d'attivazione delle parole e seleziona le possibili correzioni.
La procedura di correzione degli errori si compone dei seguenti passi:

i. ogni parola non corretta viene scomposta nei suoi n-grammi;
ii. tali n-grammi vengono utilizzati come indici nella tabella per individuare le possibili parole candidate alla correzione;
iii. i vocaboli presentati saranno tutti quelli che presentano almeno un n-gramma in comune con quelli del termine sbagliato.

Le molte varianti di questo approccio si basano in generale sulla proiezione della parola scorretta in uno spazio n-dimensionale e le parole vicine sono le possibili candidate alla correzione.
Un esempio d'implementazione di questo metodo è il programma ACUTE realizzato da Angell e al. (83).
Il sistema utilizza una tabella a tri-grammi (es. strumento = $st str tru rum ume men ent nto to$).

DeSmedt e VanBerkel (88) propongono una diversa analisi chiamata triphone analysis che permette di correggere errori nel riconoscimento del parlato.

Le prestazioni di questo sistema variano a seconda dei vocabolari utilizzati e nessun test standardizzato ha paragonato questo approccio agli altri presentati.

e. Probabilistica

Questo sistema è utilizzato per migliorare le prestazioni del precedente metodo con n-grammi.

I due indici che vengono assegnati alle possibili parole di correzione sono la **probabilità di transizione** (la probabilità che ha una determinata lettera di seguire una sequenza di caratteri) e la **probabilità di confusione** (stima della probabilità di sostituzione tra una lettera e l'altra).
I primi studi fatti hanno evidenziato come questa sola tecnica non sia sufficiente per ottenere risultati soddisfacenti. Kashyap e Oommen (84) hanno utilizzato questo metodo probabilistico per correggere parole con meno di sei caratteri (svantaggiate dal precedente approccio per n-grammi). Church e Gale (91) propongono con il loro sistema, CORRECT, un approccio ancora più complesso utilizzando quattro matrici di confusione contenenti 44 milioni di parole errate tratte da vari testi.

f. Reti neurali

L'applicazione delle reti neurali a questo campo cerca di sfruttare la versatilità che caratterizza questi sistemi per approssimare funzioni euristiche implicite: vista l'intrinseca difficoltà nel definire "regole di violazione", si cerca di far apprendere alla rete ad associare forme errate con forme presenti nel lessico attraverso cicli di addestramento in cui si mostrano "associazioni cognitivamente plausibili".
Rumelhart, Burr, Matan (Rumelhart 86; Burr 87; Matan 92) hanno adottato questo approccio in sistemi di correzione che, secondo una stima di Kukich (Kukich 92), possono raggiungere una capacità di correzione che si aggira intorno al 75% dei termini errati.
Il problema è che l'efficacia dell'approccio è strettamente dipendente dal tipo di input che si sceglie di dare in pasto alla rete (stringhe di caratteri semplici, n-grammi, sequenze fonetiche…); il problema di una correzione efficiente viene perciò semplicemente spostato, ma non risolto e una riflessione "simbolica" sulla natura del problema sembra sempre comunque fondamentale per il trattamento del problema.

(13) Alle tecniche di correzione di parole singole, che come visto spesso ottengono discreti risultati, occorre abbinare un sistema per controllare, ed eventualmente correggere, anche **espressioni dipendenti dal contesto**.
Vari autori (Thompson 80, Eastman e McLean 81; Young 91) hanno messo in evidenza che gli errori prodotti, dipendenti dal contesto, sono tra il 25% e il 50% degli errori totali, e di questi circa il 75% è di ordine sintattico.
Le due principali tecniche che affrontano il problema sono l'approccio simbolico e quello probabilistico-statistico. Il primo approccio utilizza un robusto parser e degli analizzatori morfologici e sintattici. Il secondo sistema utilizza delle tabelle di probabilità per determinare le sequenze di termini consentite. Perché questi sistemi siano robusti occorre, per il primo una solida teoria linguistica e una efficiente implementazione software, per il secondo una mole consistente di dati. Anche se in pratica soluzioni ottimali sono ancora da ricercarsi, esistono versioni interessanti di correttori accessibili a tutti:

(14) Microsoft Word XP implementa una serie di regole per la correzione di espressioni dipendenti contesto:

Regole grammaticali:

a. **Punteggiatura**: segnala gli errori sull'uso degli spazi, delle virgole, dei punti, delle parentesi e degli altri segni di punteggiatura. Esempi di errori rilevati: Dopo aver mangiato alcune mele, decise di lasciare la tavola. La decisione finale ha creato molto disagio alla popolazione.
(nessun errore è stato rivelato in queste frasi)

b. **Maiuscole**: segnala tutti gli errori relativi all'uso delle maiuscole: mancanza della maiuscola ad inizio frase, maiuscola raddoppiata, errore di battitura sulle maiuscole, maiuscole anomale. Esempi di errori rilevati: la fine dell'inverno ha portato con sé il freddo e il brutto tempo. Le scarpe di pAola sono molto costose.

c. **Genere-Numero**: segnala le discordanze in genere e in numero fra articoli o preposizioni articolate, aggettivi, pronomi e sostantivi. La regola indica anche le discordanze in genere per gli aggettivi relativi a più sostantivi e le false forme femminili o plurali. Esempi di errori rilevati: Franco ha comprato dei pantaloni e delle scarpe nuove. Le tue benevoli raccomandazioni non sono servite a nulla. Ho comprato un libri molto bello.
("anno 1966" il termine 1966 non concorda in genere-numero con le parole che lo circondano)

d. **Concordanza Soggetto-Verbo**: verifica che il soggetto e il verbo concordino tra loro; il controllo è fatto anche sui soggetti uniti da congiunzioni. La regola segnala anche l'uso di soggetti impliciti o distanti che possono rendere il testo meno comprensibile e gli errori di concordanza nei participi passati. Esempi di errori rilevati: Il cane e il gatto ha mangiato i resti del pranzo. Io speriamo di vincere un premio. Gli scolari sono uscito alcuni minuti prima del solito.

e. **Elisione-Apostrofo**: segnala gli errori relativi al troncamento e all'elisione, riconoscendone la necessità o l'uso inesatto. Esempi di errori rilevati: Non mi hanno ancora detto qual'è il tuo nuovo appartamento.

Questo uomo rappresenta una sicurezza per il suo popolo. Un tal'comportamento era del tutto inaspettato. (Elisione-apostrofo (di utenza > d'utenza))

f. **Frasi**: segnala i più comuni errori relativi alla frase e alla sua costruzione. Esempi di errori rilevati: che il pilota fosse in grado di farlo. La donna disse sarebbe andata in città.

g. **Verbi**: segnala gli errori relativi all'uso di un verbo con l'ausiliare sbagliato; questo vale anche quando un verbo è utilizzato insieme con un verbo servile (potere, volere, dovere, fare). Esempi di errori rilevati: L'aereo ha arrivato con parecchi minuti di ritardo sull'orario previsto. Io ho potuto partire per la Francia grazie all'aiuto di mio padre. Non sono voluto pranzare insieme con quelle persone.
(ma accetta tranquillamente: non sono mai e poi mai voluto pranzare insieme con quelle persone)

h. **Aggettivi**: segnala gli usi impropri degli aggettivi. Esempi di errori rilevati: lavoro molto poco in primavera. Lui non lascerà le corse, qualunque che sia la tua decisione.

i. **Articoli**: segnala gli errori causati da un uso sbagliato degli articoli: utilizzo del giusto articolo davanti a parole inizianti con consonanti particolari, uso della corretta preposizione articolata, articoli e aggettivi possessivi, articoli e sostantivi ed altri errori ancora. Esempi di errori rilevati: mia mamma vive da molti anni in una casa di campagna. Il yogurt è un alimento molto indicato per i bambini.

m. **Elementi della frase**: segnala un insieme di errori commessi con una certa frequenza e che coinvolgono diversi elementi della frase. Esempi di errori rilevati: Il figlio di mio cugino cammina ancora a gattoni. La torre di Pisa è tanto alta come bella. Ti chiedo se verrai, o meno, domani. Ho nulla in contrario. (errori rivelatori: ho mangiato tanto cioccolato come quando ero bambino > sostituire come con quanto)

n. **Preposizioni**: segnala l'esattezza nell'uso delle preposizioni insieme con sostantivi, aggettivi, pronomi, verbi ed avverbi, e segnala alcune tra le più comuni forme del parlato che sono errate nei testi scritti. Esempi di errori rilevati: Il nonno si è addormentato come al solito. La nuova macchina stampa 100 copie all'ora. Con domani inizieremo la costruzione della seconda ala dell'edificio

Regole di stile:

o. **Espressioni da evitare**: segnala le parole inutili, sprecate perché già espresse dalla parola o dal concetto precedente. Segnala le espressioni da evitare, forme d'uso comune ma spesso inutili. Un uso eccessivo di forme forzatamente ricercate, ma di dubbio gusto o grammaticalmente scorrette, appesantisce il testo senza motivo. Esempi: Ed è per questo che abbiamo deciso di modificare i piani di produzione. L'altro ieri, cioè Domenica, siamo andati al mare. Il negozio ha maggiorato i prezzi.

p. **Parole ridondanti**: segnala le parole inutili, sprecate perché già espresse dalla parola o dal concetto precedente. Questi termini non aggiungono niente al discorso e, al limite, lo rendono meno scorrevole e poco incisivo. Esempi: Per potere avere una promozione, bisogna meritarsela. Quella maionese è lievemente acidula. Le domande devono essere presentate entro e non oltre le ore 17 del 12 ottobre.

q. **Leggibilità**: segnala le forme poco scorrevoli poiché un testo contenente assonanze o forme complesse perde scioltezza e diventa difficile da leggere. Esempi: l'arciere non sapeva scegliere fra frecce rosse e frecce verdi. Il treno arrivò a Ascoli con due ore di ritardo. Il di lui cane è molto affettuoso.

r. **Termini ripetuti**: segnala la ripetizione di uno stesso termine all'interno della frase e l'utilizzo della stessa parola per iniziare frasi vicine tra loro. Esempi: La casa vicina al ponte è più bella della casa di tuo padre. Per eliminare un problema, abbiamo eliminato anche molte cose utili.

s. **Uso errato**: segnala l'uso errato di preposizioni, congiunzioni ed aggettivi utilizzati nella costruzione della frase. Si tratta spesso di forme errate diventate frequenti nella lingua parlata e per questo utilizzate anche nei testi scritti. Esempi: Questi ragazzi hanno un gran spirito d'iniziativa. Abbiamo deciso di comprarlo sia lui che io. Malgrado tutto, siete riusciti ad arrivare in tempo a scuola.

Lessico "generativo"

(15) ovvero perché ogni entrata lessicale non può essere registrata **singolarmente**

a. sarebbe **inefficiente**:

	mangi -	**sogn -**	**corr -**	**puff -**
-are / -ere	**mangi-**are	**sogn-**are	**corr-**ere	**puff-**are
-o	**mangi-**o	**sogn-**o	**corr-**o	**puff-**o
-ato	**mangi-**ato	**sogn-**ato	***corr-**ato **(**corso**)**	**puff-**ato

in Turco (lingua agglutinante) ci sarebbero circa $600x10^6$ entrate lessicali da considerare. In Finlandese 10^7

b. sarebbe **non informativo**:
   i. nessuna relazione significativa tra entrate lessicali (l'unica relazione possibile sarebbe l'ordine alfabetico, ma *casa* e *case* hanno intuitivamente una relazione più "intima" rispetto a quella tra *case* e *caso*)
   ii. non esisterebbe nessun indizio per processare in modo "particolare" ad esempio un verbo rispetto ad un nome

(16) classicamente **il lessico computazionale** era visto in funzione delle applicazioni da cui doveva essere usato. Recentemente due approcci complementari cercano di esplorare le strutture lessicali: da un lato gli studi sul **lessico mentale** cercano di fornire modelli psicologicamente plausibili di relazioni tra unità minime di significato, dall'altra si cercano di raffinare i modelli computazionali usati per creare i **database lessicali**. Da quest'ultimo punto di vista valgono le seguenti regole di condotta:

a. la rappresentazione lessicale deve essere **esplicita** ed **indipendente** dalle applicazioni che la utilizzeranno
b. la **struttura globale** delle entrate lessicale è importante almeno quanto la struttura interna delle singole parole: la sua organicità e significatività serve a rappresentare una **complessa base di conoscenza** (**ontologia**)
c. il lessico deve essere in grado di **coprire adeguatamente il suo dominio**; normalmente un lessico reale raggiunge le 400.000 entrate lessicali (approssimativamente: 5.000 entrate verbali, 30.000 nominali, 5.000 aggettivali, un miglialio di avverbiali, altrettanti termini logici, 2.000 composti e anche 300.000 nomi propri + vari termini dominio-specifici)
d. i lessici computazionali devono essere **valutabili** almeno su tre scale: i) **copertura** (sia in estensione, che in profondità, a livello di ricchezza dell'informazione); ii) **estensibilità** (deve poter essere formalmente possibile arricchire il vocabolario con termini nuovi); iii) **utilità** (stavolta valutata a livello delle singole applicazioni/elaborazioni).

due altre buone massime da tenere sempre a mente sono:

a. la **completezza** non assicura la **correttezza** (psicolinguistica e computazionale)
b. la **plausibilità psicolinguistica** non garantisce l'**efficienza computazionale** e viceversa

(17) la struttura di una **singola entrata lessicale** deve contenere le seguenti informazioni:

a. ortografiche/fonetiche (devono insomma codificare l'input nel modo più adeguato possibile)
b. morfologiche (tratti inerenti, quali plurale/singolare, massa/contabile, animato/inanimato...)
c. sintattiche (categoria grammaticale ed eventualmente la sottocategoria)
d. semantiche (sia a livello di selezione semantica, che di significato ai fini della traduzione ad esempio)

(18) cruciale è pure una buona **struttura globale del lessico**, che a sua volta permette di:

a. correlare la sottocategorizzazione con la **classe semantica** (Levin 93 propone una vasta serie di classi di alternanza cercando di dimostrare che certi comportamenti sintattici verbali, quali l'assegnazione di ruoli tematici e la selezione di complementi e aggiunti, sono prevedibili sulla base di certi tratti semantici minimalmente distintivi, quali la modificazione di stato, la causatività, la relazione tra gli elementi in azione ecc.)
b. trarre immediate **inferenze** in base all'organizzazione gerarchica degli items (part_of, member_of…)

(19) un esempio di struttura globale del lessico diverso dal consueto ordinamento alfabetico è l'organizzazione a **rete concettuale** (o **rete semantica**) degli items.

Wordnet (Miller 90) è un interessante e usatissimo esempio di rete semantica basata sui seguenti principi:

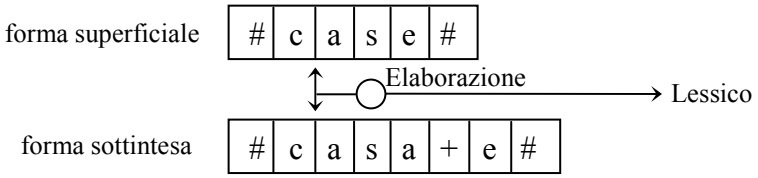
a. ogni concetto lessicale può essere **rappresentato dai suoi sinonimi** (**synsets**, ovvero altri concetti lessicali)

b. i vari synsets possono essere messi in relazione usando concetti gerarchici di **iponimia** (relazione tra un concetto generale ed uno più specifico; ad esempio "pettirosso" è un iponimo di "uccello"), **iperonimia** (specifica la relazione inversa all'iponimia) e **meronimia** (parte_di).
c. attraverso l'uso di synset distinti si può risolvere il problema della **polisemia** (*cane* = animale domestico e *cane* = parte metallica di una pistola saranno due nodi distinti di wordnet anche se si scrivono allo stesso modo)

L'analisi morfologica

(20) **obiettivo**: riconoscere una stringa ben formata di caratteri e metterla in relazione con la struttura di morfemi che la compongono; questo compito ci permette di introdurre tutti i problemi che si presenteranno nel parsing delle strutture frasali

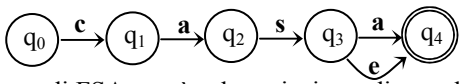
(21) **modello teorico**:



(22) **Finite-State Automata** (**FSA**): per **riconoscere** una parola come appartenente al lessico si può ricorrere agli Automi a Stati Finiti (Finite-State Automata). Formalmente definiti come una quintupla **<Q, Σ, q_0, F, δ>** dove:

Q = insieme finito e non nullo di stati
Σ = alfabeto finito e non nullo di caratteri accettabili in input
q_0 = stato iniziale, con $q_0 \in Q$
F = insieme di stati finali, con $F \in Q$
δ = insieme delle regole di transizione definite in $Q \times \Sigma$ su Q

ecco un FSA che riconosce la parola *casa* ed il suo plurale:



un insieme di FSA non è solo un insieme di macchine che permettono di riconoscere o rifiutare un elemento lessicale, ma anche di **rappresentare l'intero lessico**.

(23) **Finite-State Transducers** (**FST**, o **Trasduttori**): per **associare** una descrizione strutturale ad un elemento riconosciuto come appartenente al lessico i semplici FSA non sono più sufficienti (non esiste una memoria esterna, se non la memoria implicita data dallo stato in cui si trova l'automa, in cui "conservare" il percorso e la struttura esaminata).
Koskenniemi (83) propone un modello di morfologia a due livelli (**two-level morphology**): un livello lessicale ed uno superficiale (in modo del tutto simile a quanto proposto in (21)) che devono essere messi in una qualche relazione significativa dal punto di vista morfologico. Tale modello è implementabile utilizzando i Trasduttori a Stati Finiti (Finite-State Transducers). Un trasduttore utilizza FSA per abbinare stringhe di input a stringhe di output (teoricamente le relazioni possono essere definite anche su più livelli utilizzando output intermedi). Anche i trasduttori possono essere formalizzati come quintuple **<Q, Σ, q_0, F, δ>**, dove però sussistono alcune sostanziali differenze rispetto agli FSA:

Σ = alfabeto finito e non nullo di *caratteri complessi* accettabili in input della forma i:o dove i sono i simboli dell'alfabeto I di input e o simboli dell'alfabeto O di output. $\Sigma \subseteq I \times O$. ε (l'elemento nullo) può essere incluso sia in I che in O
δ = è definita come (q , i : o) e rappresenta la matrice di transizione che mette in relazione uno stato q di partenza e uno stato q′ di arrivo se la relazione i : o è definita. δ è quindi una relazione da $Q \times \Sigma$ su Q

i trasduttori hanno funzioni più generali degli SFA: questi ultimi descrivono un linguaggio formale definendo un insieme di stringhe ben formate, gli FST definiscono invece **relazioni** tra insiemi diversi di stringhe.
In particolare gli FST possono essere usati come **riconoscitori**, **generatori**, **traduttori**, **correlatori tra insiemi**.

alcune proprietà di cui gli FSA godono sono:
a. l'**inversione**, definita come T^{-1}, scambia le etichette di input con quelle di output
b. la **composizione**, se T_1 mappa I_1 su O_1 e T_2 è un trasduttore da I_2 ad O_2, $T_1 \circ T_2$ mappa I_1 in O_2.

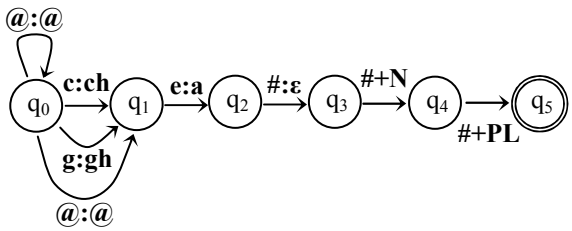
(24) Facciamo un **esempio** di FST per risolvere un problema di **morfologia flessiva**: definire un FST che descriva il fenomeno dei plurali in italiano.

i. rappresentazione del problema
esempi: casa > case; uomo > uomini; donna > donne; ago > aghi; sacco > sacchi; cane > cani;

ii. intuizioni e generalizzazioni
i nomi femminili prendono il plurale in *e*, i maschili in *i*. *c* e *g* diventano rispettivamente *ch* e *gh* al plurale.

iii. formalizzazione
caso regolare: nome maschile > @:@ c|g|@:ch|gh|@ o|e:i
nome femminile > @:@ c|g|@:ch|gh|@ a:e
caso irregolare: uomo > @:@ o:i #:n #:i

iv. implementazione
nome femminile > @:@ c|g|@:ch|gh|@ e:a #:ε #:+N #:+PL
es. case > casa +N +PL (c:c a:a s:s e:a #:ε #:+N #:+PL)



praticamente non iii può ragionevolmente essere considerato sufficiente: esistono algoritmi automatici (descritti in Kaplan e Kay 94) che mappano le regole formalizzate in relazioni a due livelli, evitando l'implementazione manuale di complessi FTS (algoritmi simili a questi stanno alla base di tutti i compilatori)

(25) Certe lingue mostrano fenomeni più problematici di quelli appena descritti. Tali fenomeni sono detti di **morfologia non-concatenativa**.
In Tagalog (un dialetto parlato nelle Filippine), ad esempio, è possibile inserire infissi nel mezzo della parola: l'infisso *um*, che marca l'agente dell'azione, può inserirsi nella parola *hingi*, che significa *prestare*, formando *h-um-ingi*.
Comune in varie lingue semitiche è altro caso di morfologia non-concatenativa, quello della morfologia a modelli (templatic morphology): una radice verbale composta da tre consonanti (CCC, ad esempio *lmd* che significa apprendere) può essere flessa inserendo schemi vocalici (CVCVC, come nel caso di *lamad*, studiò, oppure *lumad*, fu insegnato).
Il problema che questo tipo di morfologia pone è correlato in parte alla complessità computazionale, in parte alla difficoltà di usare le proprietà di concatenazione tra FST.

(26) **Problemi** incontrati:
a. **non-determinismo** (due o più percorsi possono essere innescati dallo stesso carattere allo stato q; transizioni ε)
b. **inadeguatezza** del modello per trattare fenomeni morfologici complessi
c. **ordine** di applicazione degli FSA (o delle regole a seconda dei punti di vista)

(27) **Applicazioni pratiche**: dato un contesto in cui un utente sta ricercando informazioni (web, archivio digitale) l'obiettivo di identificazione di elementi pertinenti alla ricerca effettuata è chiaramente legato alla possibilità di correlare la parola rilevante nel contesto dato. Varie limitazioni sono di solito causate dal fatto che non si dispone di un lessico on-line sufficientemente ricco. Al di là di strategie di ricerca "intelligenti" per determinare la rilevanza di un documento html rispetto alla ricerca effettuata dall'utente, l'elemento cruciale è determinare l'area tematica dei termini immessi nella query. L'approccio più usato è sicuramente quello per **keywords** combinate con operatori booleani (alberghi & Firenze).

Un'alternativa a questo approccio si chiama **stemming** e cerca di ricavare la radice (*stem*) delle parole da cercare in modo da effettuare ricerche più complete e tolleranti (es. da "alberghi & Firenze" si può generare una query (alberghi AND Firenze) OR (albergo AND Firenze)). L'algoritmo di **Porter Stemming** (Porter dal nome del suo ideatore) permette di fare proprio questo. Alla base del sistema c'è una semplice serie di FST a cascata per l'inglese del tipo:

ATIONAL -> ATE (es. relational -> relate)
ING -> ε (talking -> talk)

Krovetz (93) evidenzia un problema fondamentale di questo approccio: l'**ipergeneralizzazione** (es. organization > organ, generalization > generic, mentre **non cattura generalizzazioni** corrette quali matrices > matrix o European > Europe). Complessivamente si calcola che il vantaggio nell'uso dello stemming sussiste solo quando il corpus di documenti è relativamente piccolo ma tale sistema perde completamente senso ed efficacia ad esempio in una ricerca su Internet.

(28) Verrebbe infine da chiedersi quanto questi modelli di analisi morfologica siano **psicologicamente plausibili**, in particolare si sono cercate prove per vedere se parole come *correre*, *corre* e *ha corso*, sono tutte comprese come entrate distinte nel lessico umano senza nessuna struttura morfologica interna (**full listing hypothesis**) oppure se solo i morfemi costituenti sono compresi nel lessico umano e quando si ha accesso ad una parola come corre in realtà si ha accesso a due morfemi (*corr-* radice ed *-e* terza persona sing. presente) che poi vengono combinati tra di loro (**minimum redundancy**).

In realtà pare che nessuna di queste due ipotesi sia del tutto sostenibile:
Stanners ad al. (79), verificando **effetti di priming** nella ripetizione di parole (una parola viene riconosciuta più velocemente se vista immediatamente prima), mostrano che forme derivate come *happiness*, *happily* sembrano essere registrate separatamente dalla loro ragionevole radice *happy*, mentre invece le parole flesse regolarmente (es. la forma flessa *pouring* e la radice *pour*) sembrano non essere completamente distinte nel lessico.

Marslen-Wilson (94) raffinano il risultato precedente mostrando che in realtà anche le parole derivate, simili semanticamente alla propria radice, possono in qualche modo essere rappresentate unitariamente alla radice se sussiste una qualche **affinità semantica** (*government*, *govern*).

Infine Fromkin e Ratner (98), analizzando corpus contenenti **errori di pronuncia**, evidenziano come gli affissi possano comparire separatamente dalla loro radice (*easy enoughly* al posto di *easily enough*).
Questo suggerisce che il lessico mentale debba contenere alcune informazioni sulla struttura morfologica delle parole rappresentate.

Bibliografia essenziale:

Jurafasky & Martin (2000) *Speech & Language Processing*. Pretience Hall, NJ (**Cap. 2, 3**)
Ferrari G. (1991) *Introduzione al Natural Language Processing*. Calderini Bologna (**Cap. 1, 2**)

Approfondimenti:

- Hopcroft, Motwani & Ullman (2001) *Introduction to the automata theory, languages and computation*. Addison-Wesley. Boston
- Shank (2001) "I'm sorry Dave, I'm afraid I can't do that" in G. Stork *HAL's Legacy*. MIT Press
- Turroni Giancarlo (1994) *La correzione ortografica in un word processor*. Tesi di laurea Università di Bologna. Rel. O. Stock