

---

1 Giugno 2006

---

## **Part of Speech & Progetto**

Marco Pennacchiotti

pennacchiotti@info.uniroma2.it

Tel. 06 7259 7717

Ing.dell'Informazione, *stanza P1B-03* (nuova ala Ing.Inf, primo piano)

# Programma

- **Breve introduzione all’NLP**

- Linguaggi Naturali e Linguaggi Formali
- Complessità

- **Morfologia**

- *Teoria:* Morfologia del Linguaggio Naturale
- *Strumenti:* Automi e Trasduttori
- *Analisi Morfologica:* con automi e trasduttori

- **Part of Speech Tagging**

- *Teoria:* Le classi morfologiche
- *Strumenti a Analisi:* modelli a regole e statistici

- **Sintassi**

- *Teoria:* Sintassi del Linguaggio Naturale
- *Strumenti:* CFG
- *Analisi Sintattica:* parsing top-down, bottom-up, Early

- **Semantica**

- Lexical Semantics
- Sentence Semantics

# Analisi morfologica a due stadi

Dal livello superficiale al livello lessicale (PARSING)

Sono necessari due stadi ( → due trasduttori)

1. IDENTIFICAZIONE DEI MORFEMI: Data la parola in input sul nastro *A*, il trasduttore la divide su *B* nei morfemi costituenti (radice + affissi)
2. IDENTIFICAZIONE DELLA STRUTTURA: Dati i morfemi costituenti sul nastro *A*, il trasduttore identifica la categoria della radice e il significato degli affissi

*Livello Superficiale*



*Livello Intermedio*

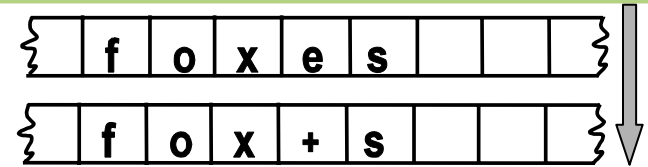


*Livello Lessicale*



PARSING

# Stadio 1: *non è così facile ...*

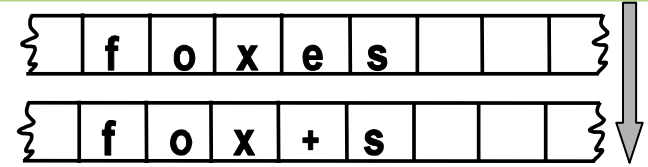


## Problemi

- Il trasduttore gestisce solo la regola della e-insertion, e non altri casi:
  - Regola y-replacement: *berries* → *berry* +s (berrie +s)
  - Regola raddoppio consonanti: *beg* → *begging*
  - *Ecc. ecc.*
- Bisogna quindi implementare **più regole ortografiche**, nello stesso trasduttore, o in trasduttori paralleli!
- **Ambiguità locale**: *foxes* produce due forme di cui solo la prima è corretta: *fox+s*, *foxe+s*, *foxes*.

# Stadio 1: Identificazione dei morfemi

## ESEMPIO: nomi singolari/plurali



## Obiettivo

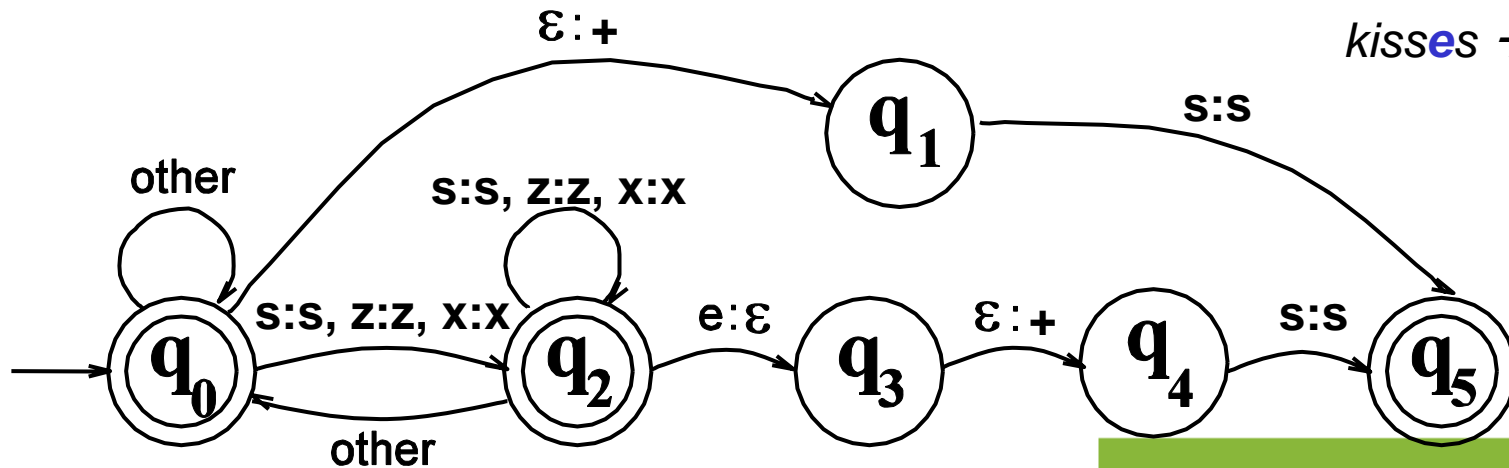
- Rappresentare con un FST le *regole ortografiche* della lingua per i nomi regolari e irregolari
- Input: cats    Output: cat+s

## Regola e-insertion

cats → cat+s

foxes → fox+s

kisses → kiss+s



# Identificazione dei morfemi

## A

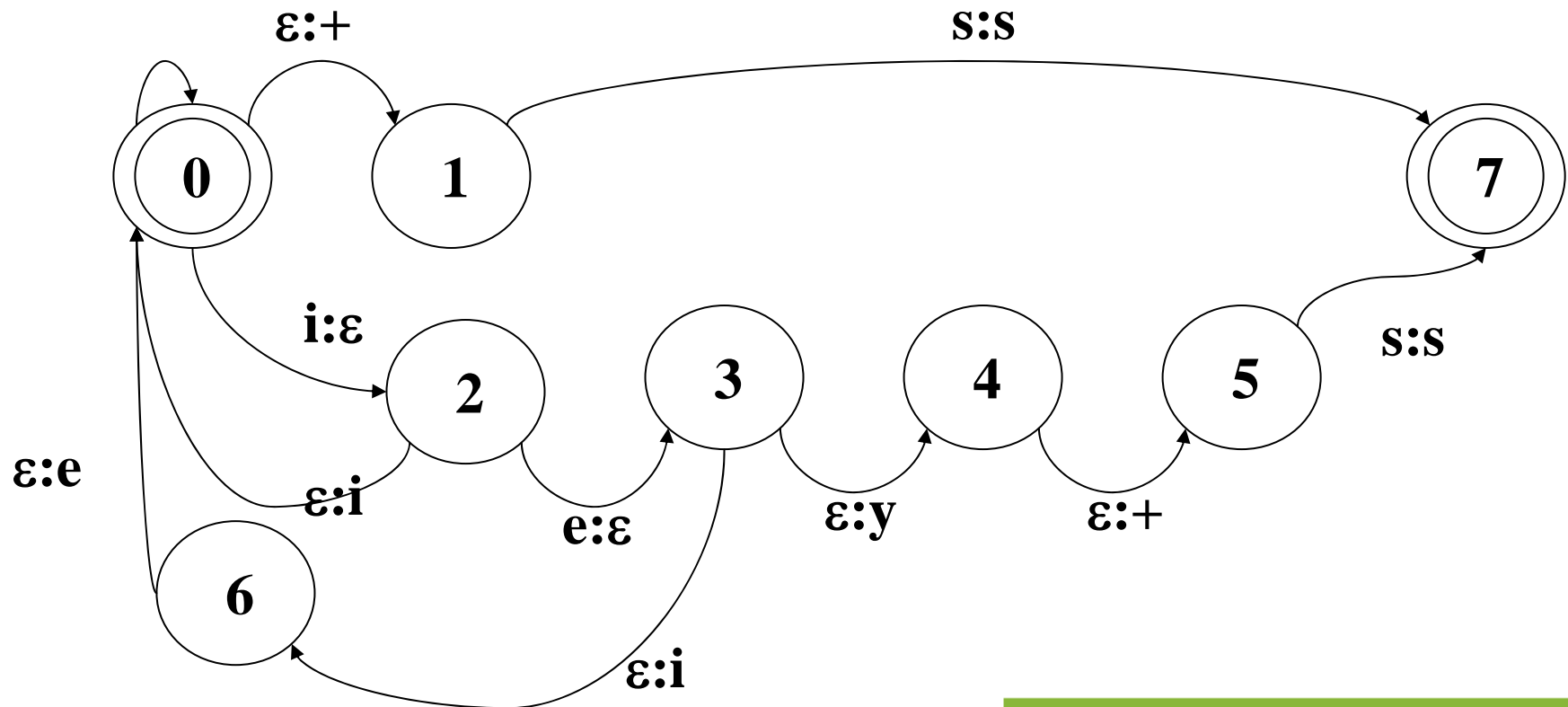
1. Scrivere l' FST che modelli la morfologia dei nomi singolari/plurali prendendo in considerazione la regola ortografica dell' *y-replacement*:
  - -y cambia in -ie prima della -s
  - ES: berry → berries

## B

1. Scrivere l' FST che modelli la morfologia dei verbi presente/passato prendendo in considerazione la regola ortografica della *k-insertion*, sapendo che:
  - In generale la forma passata si forma dal presente apponendo come suffisso la particella -ed (ES: *press* → *press-ed*)
  - La *k-insertion* prevede che ai verbi terminanti in *vocale+c* sia aggiunta la *k* (ES: *panic* → *panic + k +ed*)

# Soluzione esercizio 5

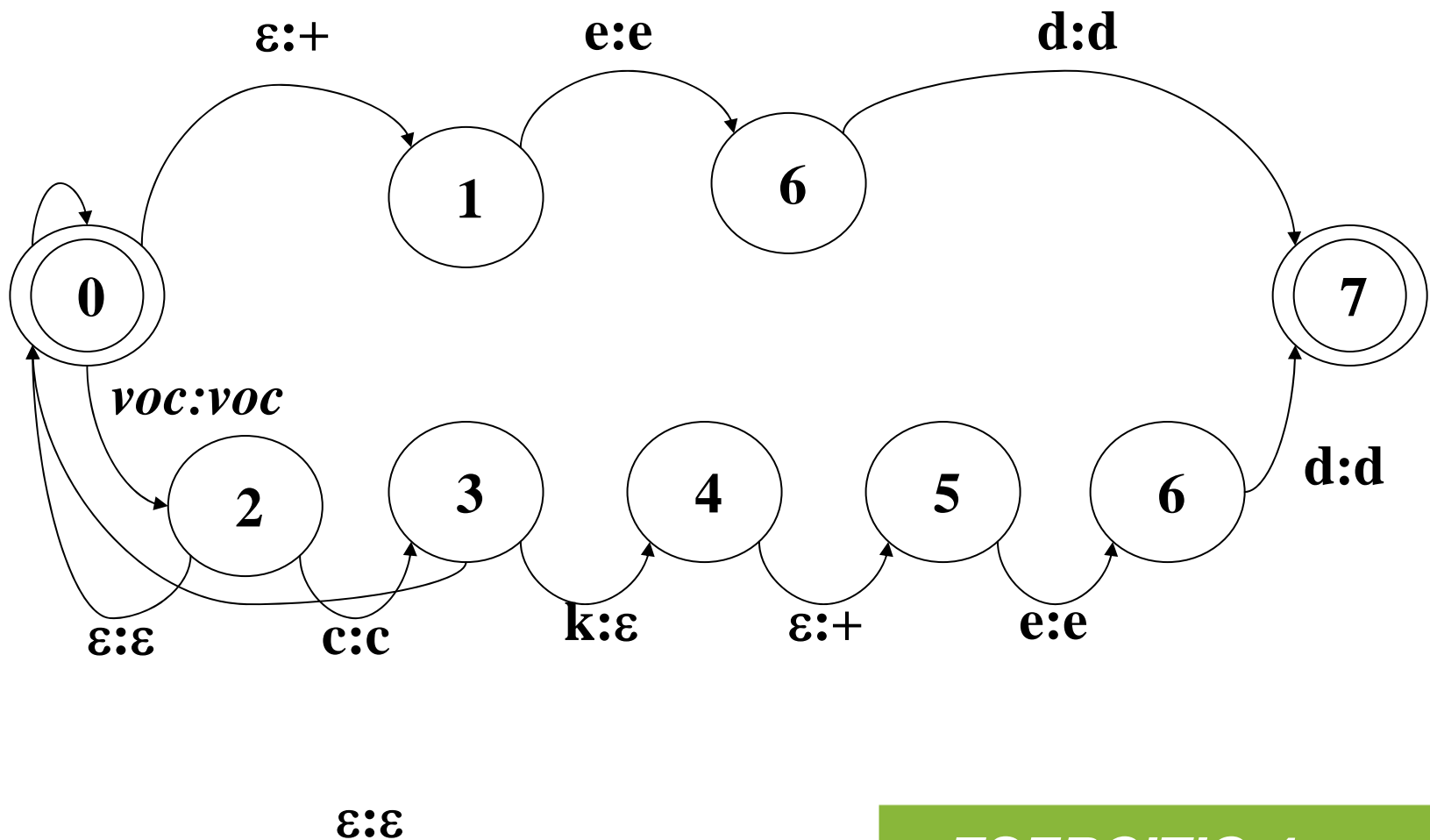
**A.1** *other*



# Soluzione esercizio 5

B.1

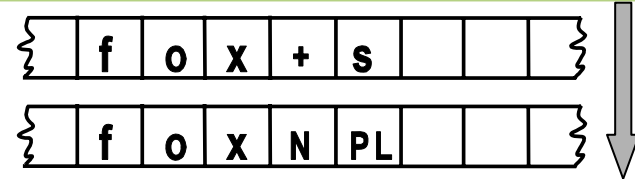
*cons:cons*





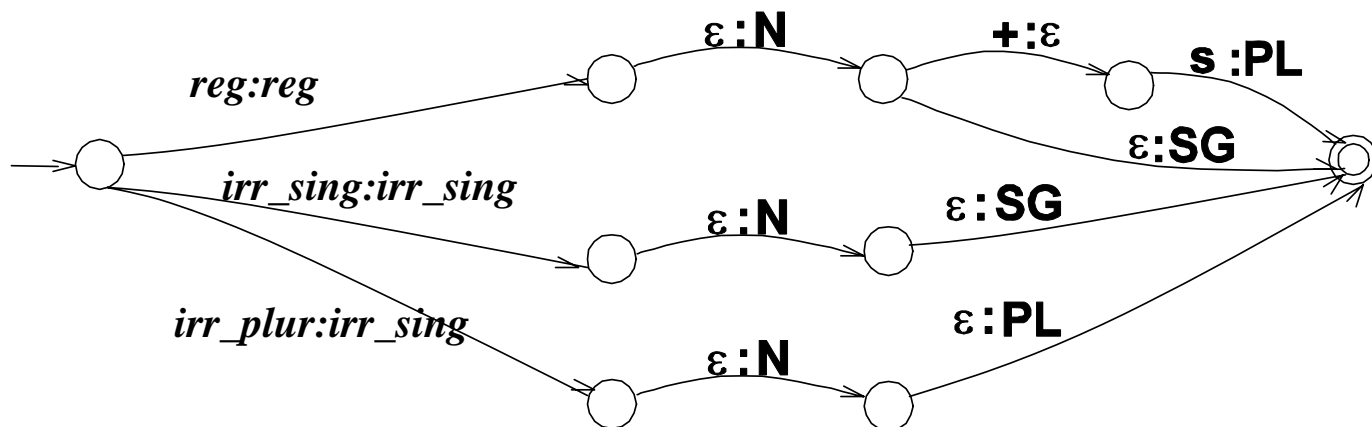
# Stadio 2: Identificazione della struttura

ESEMPIO: nomi singolari/plurali



## Obiettivo

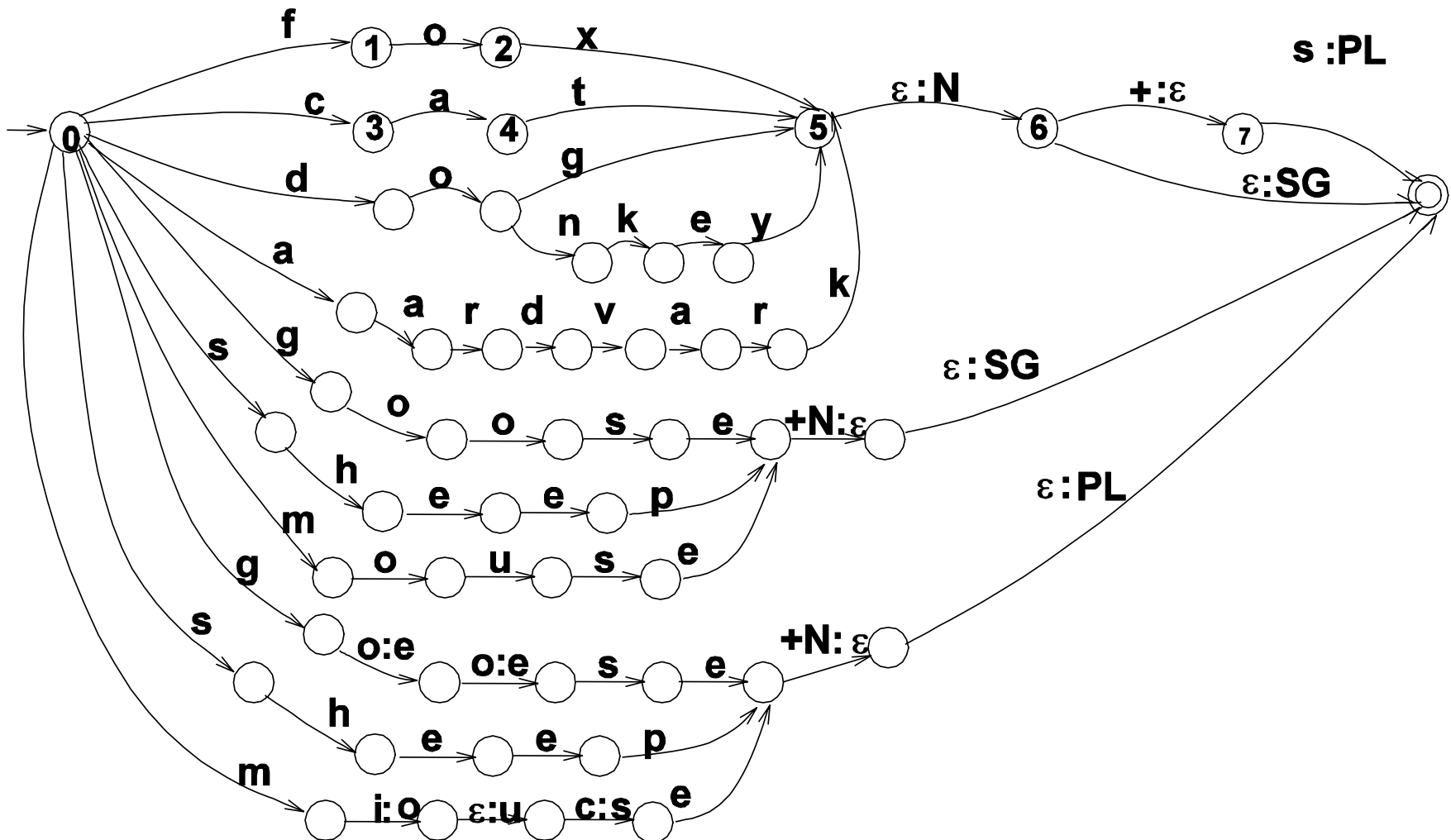
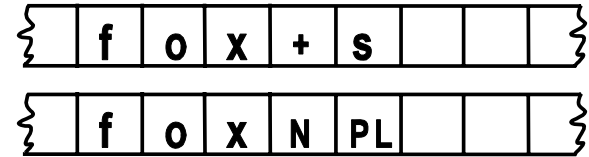
- Rappresentare con un FST le *regole morfologiche* della lingua per i nomi regolari e irregolari
- Input: *cat+s, mouse, mice*    Output: *cat N PL, mouse N SG, mouse N PL*



**Bisogna aggiungere il lessico !**

# Stadio 2: Identificazione della struttura

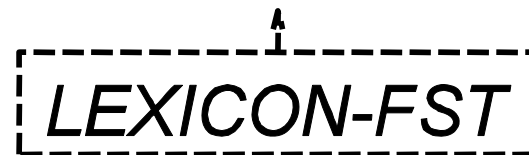
ESEMPIO: nomi singolari/plurali



# Stadio 1+2: *Combinare lessico e regole*

- E' possibile combinare i due stadi mettendo *in cascata* (in serie) i due trasduttori: l'output dell'uno sarà l'input dell'altro

*Livello Lessicale*

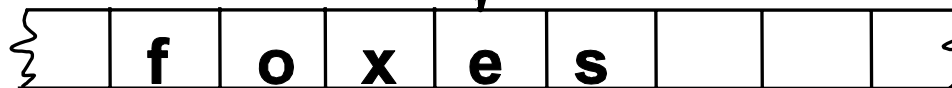


*Livello Intermedio*



} in serie o in parallelo

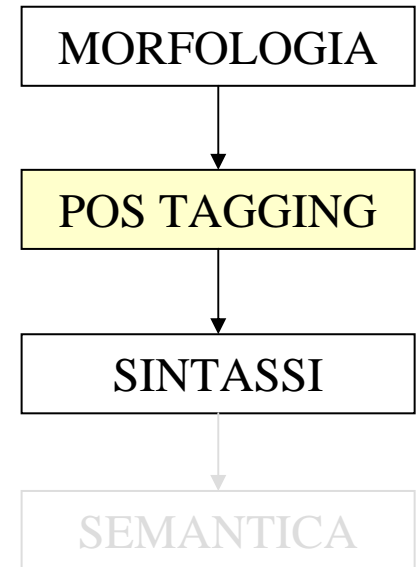
*Livello Superficiale*



- Oppure, è possibile fondere i due trasduttori, attraverso un'operazione di *intersezione*

## Part of Speech Tagging

- **Part of Speech Tagging**
  - Cos'è
  - Part of Speech
  - Part of Speech Tagging
    - » Rule-bales
    - » Stochastic
    - » Misto
  - Prestazioni



# Part Of Speech (POS)

## Part of Speech (*classi morfologiche*)

Categoria morfo-sintattiche cui una parola appartiene

### Categorie principali

- Nomi, verbi, aggettivi, avverbi, articoli, pronomi, congiunzioni ...

### Esempio

- |       |                    |                          |
|-------|--------------------|--------------------------|
| ▪ N   | <i>noun</i>        | chair, bandwidth, pacing |
| ▪ V   | <i>verb</i>        | study, debate, munch     |
| ▪ ADJ | <i>adj</i>         | purple, tall, ridiculous |
| ▪ ADV | <i>adverb</i>      | unfortunately, slowly,   |
| ▪ P   | <i>preposition</i> | of, by, to               |
| ▪ PRO | <i>pronoun</i>     | I, me, mine              |
| ▪ DET | <i>determiner</i>  | the, a, that, those      |

# Quale relazione con la morfologia ?

## ANALISI MORFOLOGICA

- Data una parola, trovare le sue interpretazioni morfologiche
- Può essere presente **ambiguità**:  
ES: talks → talk+s → talk V 3PS  
→ talk N PL

## POS TAGGING

- Data una parola, trovare la sua unica interpretazione morfologica
- Analisi morfologica + **disambiguazione**
- → metodi dell'analisi morfologica (es. FST) + algoritmi di disambiguazione

# Part Of Speech

Tradizionalmente, la definizione di una POS è basata su caratteristiche:

- **Morfologiche:** gli affissi che compongono una parola
- **Sintattiche:** il contesto sintattico in cui si trova la parola

Non sono significative di solito *caratteristiche semantiche*, sebbene le classi presentino solitamente un buon grado coerenza semantica.

## ESEMPIO (nomi)

I *nomi* in inglese hanno generalmente una forma singolare e una plurale (affisso –s)

I *nomi* in inglese sono solitamente preceduti da articoli o avere una forma possessiva (ES: *IBM's revenues*)

I *nomi* in inglese possono esprimere diverse categorie semantiche: persone, cose, astrazioni (ES. *relationship*), termini simil-verbali (ES: *pacing*)

# Classificazione delle POS

Possono essere identificate due categorie principali:

- **CLASSI APERTE:**

- Classi a cui vengono spesso aggiunte nuove parole
- Generalmente produttive
- ES: in Inglese ed Italiano sono quattro: *nomi, verbi, aggettivi, avverbi*
- *Tutte le lingue hanno almeno le classi verbo e nome (lingua universale?)*

- **CLASSI CHIUSE:**

- Classi cui appartengono un insieme relativamente statico di parole
- ES: *articoli, preposizioni, congiunzioni, parole*
- ***Function Words:*** parole grammaticalmente significative, generalmente molto corte e frequenti nel linguaggio
  - ES: *of, and, or, you ...*



# Penn Treebank tagset

Tag	Description	Example	Tag	Description	Example
CC	Coordin. Conjunction	<i>and, but, or</i>	SYM	Symbol	<i>+, %, &amp;</i>
CD	Cardinal number	<i>one, two, three</i>	TO	"to"	<i>to</i>
DT	Determiner	<i>a, the</i>	UH	Interjection	<i>ah, oops</i>
EX	Existential 'there'	<i>there</i>	VB	Verb, base form	<i>eat</i>
FW	Foreign word	<i>mea culpa</i>	VBD	Verb, past tense	<i>ate</i>
IN	Preposition/sub-conj	<i>of, in, by</i>	VBG	Verb, gerund	<i>eating</i>
JJ	Adjective	<i>yellow</i>	VBN	Verb, past participle	<i>eaten</i>
JJR	Adj., comparative	<i>btgger</i>	VBP	Verb, non-3sg pres	<i>eat</i>
JJS	Adj., superlative	<i>wildest</i>	VBZ	Verb, 3sg pres	<i>eats</i>
LS	List item marker	<i>1, 2, One</i>	WDT	Wh-determiner	<i>which, that</i>
MD	Modal	<i>can, should</i>	WP	Wh-pronoun	<i>what, who</i>
NN	Noun, sing. or mass	<i>llama</i>	WPS	Possessive wh-	<i>whose</i>
NNS	Noun, plural	<i>llamas</i>	WRB	Wh-adverb	<i>how, where</i>
NNP	Proper noun, singular	<i>IBM</i>	\$	Dollar sign	<i>\$</i>
NNPS	Proper noun, plural	<i>Carolmas</i>	#	Pound sign	<i>#</i>
PDT	Predeterminer	<i>all, both</i>	"	Left quote	<i>(' or ")</i>
POS	Possessive ending	<i>'s</i>	"	Right quote	<i>(' or ")</i>
PRP	Personal pronoun	<i>I, you, he</i>	(	Left parenthesis	<i>( [ ( { ( &lt;)</i>
PRP\$	Possessive pronoun	<i>your, one's</i>	)	Right parenthesis	<i>( ] ) } ( &gt;)</i>
RB	Adverb	<i>quickly, never</i>	,	Comma	<i>,</i>
RBR	Adverb, comparative	<i>faster</i>	.	Sentence-final punc	<i>( . ! ?)</i>
RBS	Adverb, superlative	<i>fastest</i>	:	Mid-sentence punc	<i>( : ; ... - -)</i>
RP	Particle	<i>up, off</i>			

# POS Tagging: ambiguità

Processo di assegnazione della Part of Speech ad ogni parola di un corpus (insieme di documenti)

## AMBIGUITA'

- Ogni parola dovrebbe avere un solo POS assegnato
- Molte parole sono **ambigue** (più POS tag possibili)
- Un *POS tagger* deve *disambiguare*, restituendo se possibile un solo tag:
  - Utilizzando evidenze contestuali
  - Utilizzando evidenze probabilistiche da corpora annotati

## ESEMPIO

- The **back/JJ** door
- On my **back/NN**
- Win the voters **back/RB**
- Promised to **back/VB**

## ESEMPIO

“La vecchia porta la sbarra “  
...proviamo a costruire regole per disambiguare

# POS Tagging

## Caratteristiche generali dei POS tagger:

### INPUT

- Tagset
- Dizionario con tag
- Frase da annotare

### OUTPUT

- Frase annotata

### METODI

- Rule-based: database di regole di disambiguazione
- Stochastic: risolvono le ambiguità con evidenze probabilistiche estratte da un corpus annotato (*HMM, Markov models...*)
- Misti: utilizzano regole di disambiguazione ricavate con metodi stocastici

## Part of Speech Tagging

- **Part of Speech Tagging**
  - Cos'è
  - Part of Speech
  - **Part of Speech Tagging**
    - » Rule-bales
    - » Stochastic
    - » Misto
  - Prestazioni



# POS Tagging: Rule Based

## ESEMPIO (fase 2)

**Regola 1:** Rimuovere *VBN* se è il alternativa a *VBD* e se segue “<inizio frase>*PRP*”

**Regola 2:** Rimuovere *VB* se è in alternativa a *NN* e se segue *DT*

.....

			NN		
			RB		
	<del>VBN</del>		JJ		<del>VB</del>
PRP	VBD	TO	VB	DT	NN
<i>She</i>	<i>promised</i>	<i>to</i>	<i>back</i>	<i>the</i>	<i>bill</i>

# POS Tagging: Stochastic

**Differisce dall'approccio a regole nella fase di disambiguazione:**

- Il tag corretto viene selezionato in base ad evidenze statistiche e alla teoria della probabilità
- **Approcci semplici:** *Most Frequent Tag*
- **Approcci complessi:** *HMM, Transformation-based tagging*

## **METODO MOST FREQUENT TAG**

- **IDEA:** Le parole ambigue utilizzano un tag più spesso di altri.
- **Metodo:**
  1. Creare un dizionario e annotare manualmente un corpus
  2. Per ogni parola ambigua in un nuovo testo calcolare probabilità di annotazione
  3. Assegnare il tag più probabile

# POS Tagging: Stochastic

## ESEMPIO

I/PP give/VB you/PP **a/?** pen/NN

Section/NN 381/CD **a/?**

- **Possibili tag per a:** DT NN FW
- **Qual è il più probabile ?**

## 1. Utilizzo di un corpus

- Corpus: insieme di documenti annotato manualmente con tag non ambigui (ES: *Brown Corups*, 1 mil di parole)
- Calcolare occorrenze di *a* con i diversi tag:

a/DT 21,830

a/NN 6

a/FW 3



# POS Tagging: Stochastic

## 2. Calcolo della probabilità di annotazione con un determinato tag

- ES: Qual è la probabilità che la parola abbia un certo tag?

$$P(\text{tag} \mid \text{word}) = \frac{\text{Count}(\text{word is tag})}{\text{total Count}(\text{word})}$$

- La probabilità viene stimata con delle *conte statistiche* nel corpus (ad esempio nel Brown corpus,  $P(\text{Verb}|\text{race}) = 96/98 = .98$ )
- Nell'esempio:

$$P(DT \mid a) = \frac{\text{Count}(a \text{ is } DT)}{\text{total Count}(a)} = \frac{21,830}{21,839} = 0,99996$$

$$P(NN \mid a) = \frac{\text{Count}(a \text{ is } NN)}{\text{total Count}(a)} = \frac{6}{21,839} = 0,00002$$

# POS Tagging: Stochastic

## 3. Assegnazione del tag più probabile

I/PP give/VB you/PP **a/DT** pen/NN

Section/NN 381/CD **a/DT** ← **ERRATO (NM)**

### LIMITI

- Annota bene in molti casi
- Nei casi più rari sbaglia sempre
- Per aumentare le prestazioni è necessario prendere in considerazione altre informazioni.
  - Ad esempio: guardare il tag della parola precedente e successiva risolverebbe il caso precedente
  - Implementare automaticamente in versione probabilistica le regole dei sistemi rule-based (*learning*)
- **Approcci complessi:** *HMM, Transformation-based tagging*

- **METODO MISTO (rule based + stochastic)**
  - **RULE BASED:** La scelta del tag è effettuata usando *regole contestuali (contextual rules)* e *regole lessicali (lexical rules)*
  - **STOCHASTIC:** Le regole sono apprese automaticamente con tecniche di Machine Learning
- CONOSCENZA PREGRESSA NECESSARIA:
  - *Lessico:* per assegnare i tag
  - *Corpus annotato:* per l'apprendimento delle regole
- Informazioni sul TBL (detto anche *Brill tagger*):
  - Download: <http://www.cs.jhu.edu/~brill/>
  - Articolo di riferimento  
<http://ai-nlp.info.uniroma2.it/pennacchiotti/teaching/CompLing95.pdf>
  - Brill-based tagger on line (*CST*)  
[http://www.cst.dk/online/pos\\_tagger/uk/index.html](http://www.cst.dk/online/pos_tagger/uk/index.html)

1. Prima annotazione del testo. Per ogni parola:
  - a. Se è presente nel lessico, scegli il tag migliore utilizzando la strategia *Most Frequent Tag*
  - b. Altrimenti utilizza regole lessicali (*lexical rules*) per derivare il possibile tag
2. Applicazione al testo delle *regole contestuali*. Per ogni parola:
  - Applica le regole dall'alto verso il basso
  - Regole generali si trovano in alto, specifiche in basso

## ESEMPIO

1a. Section/NN 381 a/DT mistake/NN was/VBD found/VBN

1b.LR Section/NN 381/CD a/DT mistake/NN was/VBD found/VBN

2.CR1 Section/NN 381/CD a/NN mistake/NN was/VBD found/VBN

2.CR2 Section/NN 381/CD a/DT mistake/NN was/VBD found/VBN

LR1 If CIFRA then CD

CR1 If PREC is CD then NN

CR2 If PREC is CD and  
SUCC is NN then DT

- Come vengono apprese le regole contestuali ?

- Corpus annotato manualmente (*gold standard*)

1. Prima annotazione del corpus con metodo *Most Frequent Tag*

**Repeat**

2. Applicazione al testo di particolari template di trasformazione, istanziandoli a turno su tutti i tag presenti nel Tagset

- ES: Cambia *tag1* in *tag2* se PREC ha *tag3*
- ES: Cambia *tag1* in *tag2* se SUCC e SUCC+1 hanno *tag3* e *tag4*

3. Scelta del template che migliora al massimo l'accuratezza rispetto al gold standard

**Until** miglioramento accuratezza < threshold

- **Regole Contestuali :**
  - cambiano il *tag* di una parola in base al tag delle parole circostanti
- *Es:*  
***NN VB PREVTAG TO***  
cambia da NN a VB se tag precedente è TO  
*to/TO run/NN → to/TO run/VB*

***Esempi di abbreviazioni utilizzate:***

1. PREV --- previous(preceding)
2. PREVTAG --- preceding word is tagged
3. PREV1OR2TAG --- one of the two preceding words is tagged
4. PREV1OR2OR3TAG --- one of the three preceding words is tagged
5. WDAND2AFT --- the current word is x and the word two after is y
6. PREV1OR2WD --- one of the two preceding words is
7. NEXT1OR2TAG --- one of the two following words is tagged
8. NEXTTAG --- following word is tagged
9. NEXTWD --- following word is
10. WDNEXTTAG --- the current word is x and the following word is tagged z
11. SURROUNDTAG --- the preceding word is tagged x and the following word is tagged y
12. PREVBIGRAM --- the two preceding words are tagged
13. CURWD --- the current word is

- **Regole lessicali:**
  - Utilizzate quando una parola non è nel lessico
  - Cercano di *indovinare* la classe della parola in base a informazioni morfologiche
  - Riducono la lunghezza del lessico
- *Es:*

## **NN s fhasuf 1 NNS**

change the tag of an unknown word from NN to NNS if it has suffix -s  
*webpages/NN to webpages/NNS*

## **NN . fchar CD**

change the tag of an unknown word from NN to CD if it has character '.'  
*3.5/NN to 3.5/CD*

## **NN ing fhasuf 3 VBG**

change the tag of an unknown word from NN to VBG if it has suffix -ing  
*tagging/NN to tagging/VBG*

- Debolezze:

- Alcune regole possono essere corrette in alcuni casi, e produrre errori in altri:

*NN VBD PREV1AND2TAG DT NN*

*The/DT dog/NN **bit/NN** the/DT boy/NN yesterday/NN ./.*

OK!

*The/DT dog/NN **bite/NN** hurts/VB*

ERR!

- Alcune espressioni linguistiche non sono catturabili da regole:

*The/DT raft/NN **floated/VBD** down/RP the/DT river/NN sank/VBD ./.*

*(floated* dovrebbe essere *VBN*, non *VBD*)



## Part of Speech Tagging

- **Part of Speech Tagging**
  - Cos'è
  - Part of Speech
  - Part of Speech Tagging
    - » Rule-bales
    - » Stochastic
    - » Misto
  - **Prestazioni**

# POS Tagging: prestazioni

## Come misurare le prestazioni di un POS tagger

- **GOLD STANDARD:** annotazione portata a termine da un umano. Viene usato per valutare le prestazioni generali del sistema:
  - **ES:** accuratezza 0.96 (96% di tag corretti rispetto al gold std.)
- **BASELINE:** serve a definire la difficoltà del task, comparando le prestazioni del sistema con quelle di un sistema “basilare”
  - Per il POS tagging: *Most Frequent Tag System*
- **CEILING:** serve a definire la difficoltà del task, comparando le prestazioni del sistema con quelle di un sistema “ottimo”.
  - Misurato come *agreement* (accordo di annotazione) tra più annotatori in riferimento al gold std.

### POS TAGGING

Prestazioni medie: 0.96

Ceiling: 0.97

Prestazioni TBL: 0.97 (200 regole)

Baseline: 0.90

# POS Tagging: Rule Based

Dati i seguenti frammenti:

- a. It is a nice night.*
- b. This crap game is over a garage in Fifty-second Street*
- c. I am sitting in Mindy's restaurant putting on the gefillte fish, which is a dish I am very fond of.*

1. Per ogni parola trovare su un dizionario inglese i possibili POS tag in accordo con il tagset del Penn TreeBank (fase 1)
2. Effettuare manualmente il POS tagging, lasciando i POS tag ambigui (fase 1)
3. Disambiguare (fase 2) scrivendo manualmente delle regole di disambiguazione (informalmente)

# Chaos: Pos Tagging IT

## CHAOS POS tagger per l'italiano

[/data/KB/it/POStagger]

- Basato su Brill-tagger
  - Lessico di 12.000 parole con POS tag ordinati per frequenza (LEXICON)
  - Training delle regole effettuato sul corpus *Sole24 ore - Sital*:
    - 126 regole contestuali (CONTEXTUALRULEFILE)
    - 181 regole lessicali (LEXICALRULEFILE)
- Il Most Frequent Tag di una parola è il primo della lista:  
ES: *logica* **NCS** AGS

# Chaos: Pos Tagging IT

## CHAOS tag-set per l'Italiano

POS TAG	SPIEGAZIONE
AGS	Aggettivo Singolare
AGP	Aggettivo Plurale
ADS	Aggettivo Determinativo Singolare
ADP	Aggettivo Determinativo Plurale
AGI	Aggettivo Interrogativo
NUM	Numero
ARS	Articolo Singolare
ARP	Articolo Plurale
AVV	Avverbio
COA	Congiunzione Avverbiale
CPU	Congiunzione Punto
COP	Congiunzione Parentesi
COS	Congiunzione Subordinativa
CO	Congiunzione
DAT	Data
PRN	Pronome Interrogativo
PSG	Pronome Singolare
PPL	Pronome Plurale
PRR	Pronome Relativo
PR	Pronome
NCS	Nome Comune Singolare
NCP	Nome Comune Plurale
NC	Nome Comune

POS TAG	SPIEGAZIONE
NPR	Nome Proprio
PSE	Preposizione Semplice
PAS	Preposizione Articolata Singol
PAP	Preposizione Articolata Plurale
PIM	Preposizione Impropria
VX	Verbo Ausiliare
VFT	Verbo Finito Transitivo
VFI	Verbo Finito Intransitivo
VNT	Verbo NonFinito Transitivo
VNI	Verbo NonFinito Intransitivo
VNP	Verbo NonFinito TransPartPass
VIP	Verbo NonFinito IntranPartPass
VTR	Verbo Transitivo
VIN	Verbo Intransitivo
SYM	Simbolo

# POS Tagging: Chaos

Verificare quale interpretazioni di POS tagging Chaos produrrebbe avendo in input le seguenti frasi:

- a. *“La vecchia porta la sbarra”*
- b. *“I maestri di Platone furono critici”*

Analizzando il processi nelle sue diverse fasi:

1. Ricerca nel lessico delle parole
2. Utilizzo delle regole lessicali per le parole non presenti nel lessico
3. Disambiguazione: scelta del POS tag più frequente
4. Utilizzo delle regole contestuali

# PROGETTO

- Il progetto consisterà nel simulare su piccola scala un classico *esperimento* di NLP:
  - Definizione e **setup** dell'esperimento e creazione del corpus
  - Implementazione di un semplice **sistema** di analisi semantica (o altro)
  - Misurazione delle **prestazioni** del sistema sul corpus
- **Fasi del progetto**
  1. Creazione del corpus annotato:
    - Eseguire e correggere Chaos sull'analisi:
      - Morfologica
      - POS
      - Sintattica
    - Misurare le performance di Chaos
  2. Implementazione del sistema e verifica delle prestazioni sul corpus annotato.

# PROGETTO

- Il progetto consisterà nel simulare su piccola scala un classico *esperimento* di NLP:
  - Definizione e **setup** dell'esperimento e creazione del corpus
  - Implementazione di un semplice **sistema** di analisi semantica (o altro)
  - Misurazione delle **prestazioni** del sistema sul corpus
- **Fasi del progetto**
  1. Creazione del corpus annotato:
    - Eseguire e correggere Chaos sull'analisi:
      - Morfologica
      - POS
      - Sintattica
    - Misurare le performance di Chaos
  2. Implementazione del sistema e verifica delle prestazioni sul corpus annotato.



# Come si effettua un esperimento

In NLP un esperimento ha generalmente lo scopo di verificare quanto un modello o un sistema è *desiderabile* (efficace) nel portare a termine il proprio task, e di permettere una comparazione con altri modelli/sistemi.

- Tre punti fondamentali:

1. Cosa misurare (DEFINIZIONE DEL TASK)
2. Come misurare il sistema (PRESTAZIONI)
3. Dove misurare il sistema (CORPUS)
4. Come giudicare il sistema (COMPARAZIONE)

# Come si effettua un esperimento

## 1. Definizione del task:

- Il task deve essere ben definito ed essere “*misurabile*” (deve essere possibile misurarne le *prestazioni*)
- Generalmente un task di NLP può essere ridotto ad un **problema di classificazione** (eventualmente preceduto da un **problema di segmentazione**), in cui le classi sono *discrete* e in *numero finito*
  - Nel caso più semplice vi sono solo due classi (*classificazione binaria*)
  - In casi più complessi vi sono più classi (*multi-classificazione*)
- Il sistema (s) deve *predire* la classificazione *il più possibile corretta*
- Un *gold standard (gs)* manuale viene utilizzato per valutare la correttezza

### Esempio

- *Pos-tagging*
  - Multi-classe: assegnazione del POS tag ad ogni parola; le classi sono i differenti POS (NN, NNP, VBD...)
  - Binario: assegnazione del POS tag *NN* alle parole; classe 1 (assegna NN) e 0 (non assegna NN)

# Come si effettua un esperimento

## 2. Prestazioni (caso binario)

- Le prestazioni del sistema vengono misurate generalmente utilizzando una o più delle seguenti misure:
  - Precision, Recall, F-Measure**
    - Precision*: quanta dell'informazione giudicata di classe 1 dal sistema è effettivamente di classe 1
    - Recall*: quanta informazione di classe 1 il sistema ha classificato correttamente
    - F-Measure*: misura che tiene in considerazione sia la Precisione che la Recall (media armonica tra Precisione e Recall)
  - Accuracy**
    - Quanta dell'informazione è stata classificata correttamente dal sistema

### Confusion Matrix

		sistema	
		1	0
gs	1	a	b
	0	c	d

Classificazioni corrette (a, d)  
Classificazioni errate (b, c)

$$\text{Accuracy} = (a+d) / (a+b+c+d)$$

$$\text{Precision} = a / (a+c)$$

$$\text{Recall} = a / (a+b)$$

$$\text{F-Measure} = (2 * \text{Prec} * \text{Rec}) / (\text{Prec} + \text{Rec})$$

# Come si effettua un esperimento

- Quando utilizzare:
  - **Accuracy**
    - Quando ho problemi multiclasse, e tutte le classi sono di interesse (es. pos-tagging multiclasse)
    - Quando ho problemi binari, e le due classi sono bilanciate ed entrambe interessanti
  - **Precision, Recall, F-Measure**
    - Nei casi in cui c'è almeno una classe non interessante (es. pos-tagging binario)

## Esempio

- *Pos-tagging binario*
  - *Classe 1: interessante , Classe 2: non interessante*
  - *Parole classe 1 (NM) in un certo testo sono il 5% (secondo il gs)*
  - *Ammettiamo che il sistema classifichi tutte le parole in classe 0*
  - *Tale sistema è ovviamente inutile (infatti  $Precision=0$ ,  $Recall=0$ )*
  - *Eppure:  $Accuracy=95\%$  !!!*

# Come si effettua un esperimento

## ■ 3. Corpus

- Il sistema viene misurato su un insieme di testi, detto *corpus*. Un corpus può essere:
  - *Generico* : i testi riguardano diversi argomenti (politica, sport, cronaca, etc...)
  - *Specifico/di dominio* : i testi riguardano un argomento specifico
- Il corpus può essere preventivamente annotato manualmente con informazioni morfologiche, POS, sintattiche, eccetera
- Alcuni corpora:
  - Brown Corpus (EN): generico, 1.000.000 parole,
  - British National Corpus (EN): generico, 100.000.000 parole

# Come si effettua un esperimento

## ■ 4. Come giudicare le prestazioni

### COMPARAZIONE

- Un sistema viene generalmente comparato con altri sistemi, sullo stesso task e sullo stesso corpus.
  - *baseline*: sistema “stupido” che adotta la tecnica più semplice per risolvere il task:
    - Es: posTagging multiclasse → assegna il tag utilizzando *Most Frequent Tag*
    - Es: posTagging binario → assegna il tag NN a tutte le parole

# Come si effettua un esperimento

## ■ 4. Come giudicare le prestazioni

### COMPLESSITA'

- Un task può essere più o meno complesso:
  - Basse prestazioni non implicano necessariamente un pessimo sistema se il task è complesso
  - Alte prestazioni non implicano necessariamente un ottimo sistema se il task è semplice
  - Stima della **complessità del task**:
    1. Baseline alta
    2. un task è complesso se è difficilmente solvibile da un essere umano: ovvero, se due persone (*annotators*) producono classificazioni differenti sul task
      - **Inter-annotator agreement**: quanto due o più annotatori concordano sulla classificazione
      - **Kappa**: misura l'inter annotator agreement:
$$k = \frac{P(A) - P(E)}{1 - P(E)}$$

P(A) = agreement osservato  
P(E) = agreement atteso (dovuto al caso)

# Come si effettua un esperimento

## ■ Stima del kappa

$$k = \frac{P(A) - P(E)}{1 - P(E)}$$

$$P(A) \approx \frac{(a + d)}{(a + b + c + d)}$$

		Annotator 1	
		1	0
Annotator 2	1	a	b
	0	c	d

$$P(E) \approx \frac{(a+b)(a+c)}{(a+b+c+d)} + \frac{(c+d)(b+d)}{(a+b+c+d)}$$

Probabilità che annotatore 2 annoti nella classe 1

Probabilità che i due annotatori siano in agreement sulla classe 1

Probabilità che i due annotatori siano in agreement sulla classe 0



# Come si effettua un esperimento

## Valori di agreement

- Più alto è il valore del kappa, più accordo c'è tra gli annotatori → task più semplice

Interpretazione del Kappa	
<i>kappa</i>	<i>Interpretation</i>
<0	No agreement
0.0-0.19	Poor agreement
0.20-0.39	Fair agreement
0.40-0.59	Moderate agreement
0.60-0.79	Substantial agreement
0.80-1.00	Almost perfect agreement

- Alcuni task di NLP:

- **POS-tagging multiclasse** (task poco complesso)

- Baseline accuracy: 0.90
- kappa : 0.97
- Brill tagger accuracy : 0.97

- **Textual entailment** (task complesso)

- Baseline accuracy: 0.50
- kappa : 0.60
- State-of-the-art accuracy 0.70