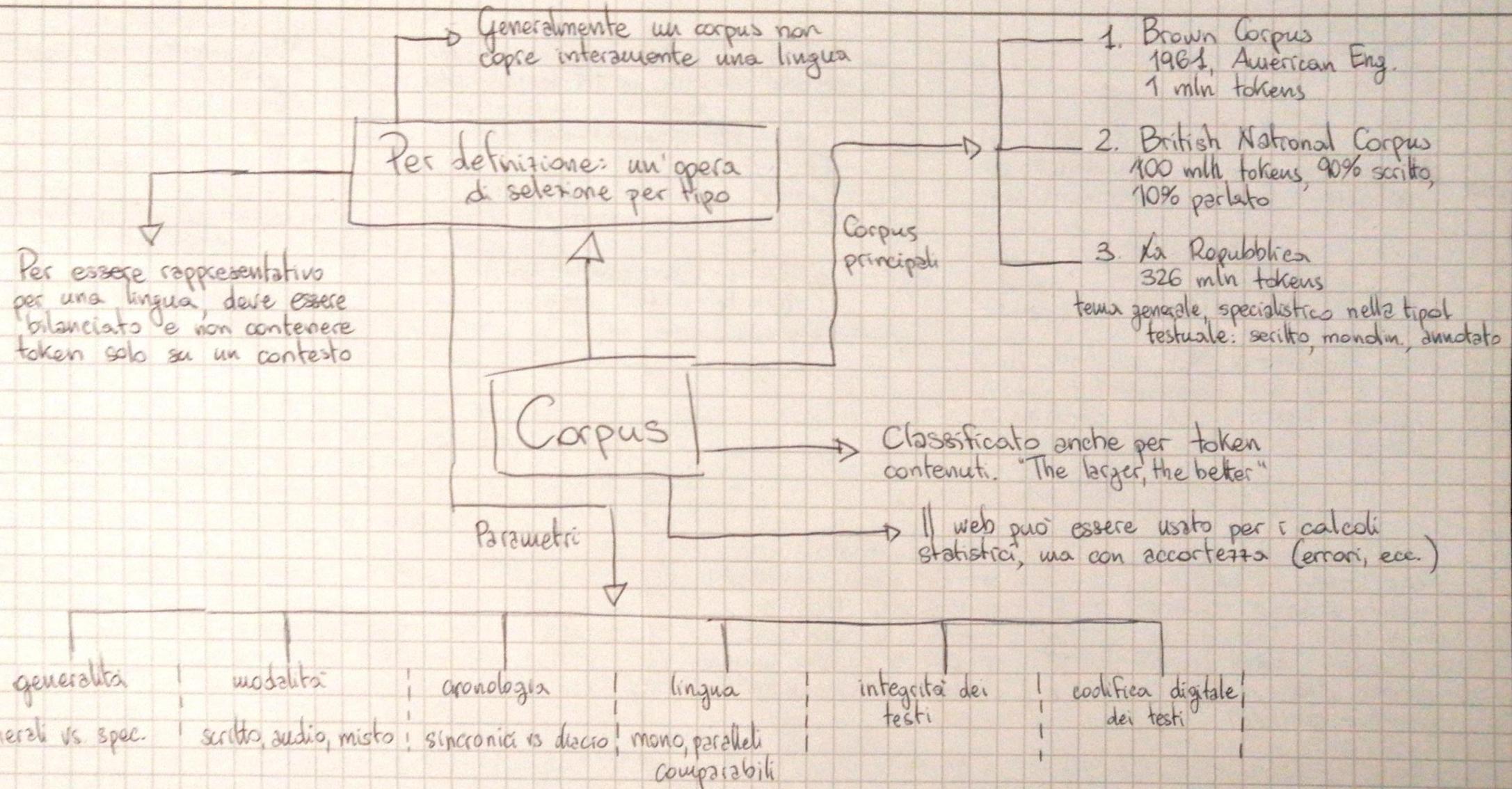


Omnia



Codifica di un testo

Un testo, per essere analizzato da un "pc" deve essere in Machine Readable Form

analizzare

Per codificare un testo, si effettuano dei passaggi:

- test cleaning: pulizia da errori vari
- verifica dei char: si usa una codifica uniforme
- normalizzazione: si rimuovono maiuscole (problemi di ambig.)
- tokenizzazione: word-segmentation per lingue (segn. (cinese, giappo, ...))

Analisi statistica del testo

Una volta tokenizzato "correttamente" il testo una prima operazione di analisi consiste nel contare le frequenze assolute delle parole. La somma di esse è uguale a $|C| = f(v_1) + \dots + f(v_n)$

Dalle frequenze assolute, ricaviamo la frequenza relativa dividendo $f(v_i) \Rightarrow F(v_i) = f(v_i)/|C| \rightarrow F(v_i) \cdot 100$ ci dà con quale percentuale quel token compone $|C|$

↳ frequenza relativa cumulata da le frequenze totali su parte di $|C|$

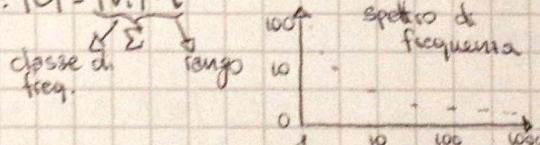
Codifica di livello 0:

effettua una semplice rappresent. binaria del testo (ASCII - UTF8)
 * 1. ASCII
 2. ISO-LATIN
 3. UTF
 4. ...
 ↓
 quantità maggiore di bit utilizz.

Codifica di livello 1:

venzono aggiunti, al liv 0, dei "metadati" informativi tramite linguaggi di mark-up (XML)

Classi di frequenza: si raggruppano i type con la stessa freq.: $|C| = |V_i| \cdot t$



Per avere $|C| = |V|$
 ogni elemento di V deve avere $f(v) = 1 \Rightarrow$ happen legge della

$|C|$ = lunghezza intera del corpus (tokens)

$|V|$ = lunghezza del vocabolario (types)

Token-Type Ratio \rightarrow indice di ricchezza lessicale

$$TTR = \frac{|V|}{|C|}$$

Generalmente è un valore $0 < TTR \leq 1$

$$|C| = 0$$

$$|C| = |V|$$

Anche se i vocabolari di due testi coincidono al 10%, i testi possono presentare anche un 70% di token uguali

→ parole "vuote" o grammaticali, sono articoli, congiuntioni, ecc e sono le più frequenti

→ parole "piene" o lessicali, sono nomi, verbi, aggettivi, meno frequenti e compongono il senso del testo.

Esistono sempre parole:

- pochi type, ma alta frequenza (gramm.)
- molti type, ma bassa frequenza (lessicali) (LNRE, Large Number of Rare Events)

L'andamento può variare al crescere del vocabolario, ossia un segno di discontinuità nel testo

d'andamento tendenziale di f al crescere dei campioni è una funzione non decrescente $f(n) \leq f(n+1)$

vocabolario aperto
tendenzialmente un testo non copre il 100% di una lingua
dato spazioso
la maggior parte delle parole hanno freq diverse

Analisi statistica del testo

La frequenza media delle parole in C deriva dalla definizione di media aritmetica

$$\bar{f} = \frac{f_{v_1} + \dots + f_{v_n}}{n} \Rightarrow \bar{f}(1c1) = \frac{|C|}{|Vc1|}$$

è l'indice inverso della TTR

Il range r_v di una parola è la posizione che essa occupa in base alla sua frequenza dal maggiore al minore

da legge di Zipf dice che la frequenza di una parola è inversamente proporzionale al suo range

$$f(z) = C/z^a$$

- $f(z)$: frequenza della parola di rango z
- C : costante corrispondente alla $f(1)$
- z : il range
- a : una variabile approssimata a 1

$$f(1) = C, f(2) = C/2, f(n) = C/n$$

In generale, al crescere del range, lo scarto $(C/n) - (C/(n+1))$ tra le frequenze di due parole, di ranghi adiacenti, diminuisce

{ da legge di Zipf prevede un decremento progressivo della frequenza di una parola proporzionale al suo aumento di range

↳ Applicazione logaritmica: rende un rapporto costante in differenti costanti

$$\log f(z) = \log C/z^a = \log C - a \log z \Rightarrow q + mx$$

appiattisce la linea dello spettro di frequenze

Un modello probabilistico di un evento determina la probabilità che si verifichi oppure no.
(evento aleatorio)

Analisi probabilistica del testo

Un corpus di testi si usa per calcolare le statistiche sulle distribuzioni delle strutture linguistiche e costruire un modello che assegna una probabilità a eventi.

Natural Language Processing

→ maggiore robustezza in mancanza di info

→ capacità di generalizzazione

→ approccio induttivo ed empirico, stimando a partire dai training corpus

Probabilità, un numero che indica il grado di incertezza al verificarsi di un evento

$$P(A) = \frac{|A|}{\Omega} \text{ dove } |A| \text{ è la freq. relativa di un evento e } \Omega \text{ lo spazio campione}$$

La stima frequentista della probabilità viene chiamata Maximum Likelihood Estimation (MLE) dei parametri di un modello probabilistico. Tende a massimizzare la probabilità di v nel linguaggio

Nel caso della L.C. lo traduciamo come

$$P(v) \approx \frac{f(v)}{|C|} \Rightarrow f(v) = |A| \text{ e } |C| = \Omega$$

Il fenomeno del data sparseness tende a non rendere affidabile la stima probabilistica in caso di frequenze estremamente basse.

Per risolvere il problema del data sparseness si usa la tecnica dello smoothing, smussamento, o anche detta Add-one smoothing.

$\rightarrow P(A) \geq 0 \quad \forall A$, non può essere negativo

$* P(A \cup B) = P(A) + P(B) \text{ se } A \cap B = \emptyset \text{ (regola somma)}$

$\rightarrow P(\Omega) = 1$ (certezza)

* Il postulato della somma, intuitivamente, implica quello di certezza

Deriva dalla legge di Laplace e permette la stima di eventi non presenti nel corpus.

$$P(v) = \frac{f(v)}{|C|} \Rightarrow P_s(v) = \frac{f(v)+1}{|C|+|V|}$$

Probabilità congiuntuale

$P(A \cap B) \text{ o } P(A, B)$ è la probabilità che due eventi si verifichino assieme.

Si calcola su eventi indipendenti, ossia il verificarsi di A non implica il verificarsi di B. In questo caso vale che $P(A, B) = P(A) * P(B)$

Probabilità condizionata

$P(B|A)$ è la probabilità che si verifichi B se ~~è stata~~ A è avvenuta

Si applica nel caso di eventi dipendenti, sapendo che è avvenuto A, questo altera la probabilità di B: $P(B) \neq P(B|A)$

Per calcolarla, si utilizzano le frequenze dei bigrammi
 $P(B|A) = f(A, B) / f(A, v_i)$. Poiché $f(A, v_i) = f(A)$

$$P(B|A) = \frac{f(A, B)}{f(A)}$$

In generale vale che, se $P(A) \neq 0$, allora $P(B|A) = P(A \cap B) / P(A)$ e, per il prodotto, vale $P(A \cap B) = P(A) * P(B|A)$.
Se A e B sono indipendenti, $P(B|A) = P(B)$
allora $P(A \cap B) = P(A) * P(B|A) \Rightarrow P(B) * P(A) * P(B)$

Language Models

Assegnano una probabilità a una sequenza di parole $P(v_1, \dots, v_n)$
Vengono chiamati n-gram models

Uno dei campi in cui si applicano gli n-gram models è quello della word prediction: con quale probabilità, data una sequenza v_1, \dots, v_n , avremo v_{n+1} : $P(v_{n+1}|v_1, \dots, v_n)$

Dato il rapporto fra prob. cong. e cond., equivale a trovare la parola che massimizza il valore di $P(v_1, \dots, v_n, v_{n+1})$

Le probabilità di sequenze di n-grammi si possono misurare tramite le frequenze degli n-grammi:

$$P(A_1, A_2, \dots, A_{n+1}) \approx \frac{F(A_1, \dots, A_n)}{F(A_1, \dots, A_{n-1})}$$

Per stimare la probabilità di n-grammi, si utilizza(va) la tecnica chain rule:

$$P(A_1, \dots, A_N) = P(A_1) * P(A_2|A_1) * \dots * P(A_N|A_1, \dots, A_{N-1})$$

Ci permette di calcolare la probabilità di sequenze di caratteri, parole o frasi.

Non si riconoscono le frasi anagrafiche.

Si può lavorare su sequenze brevi, o avere sequenze molto lunghe che complicherebbero i calcoli a lungo andare.

Trattandosi di un prodotto, basta solo una $P=0$ per annullare tutto.

Modello di Markov

Questo modello probabilistico semplifica i calcoli su sequenze di parole, limitandoli a semplici n-grammi che non invalideranno il valore finale.
Se vogliamo la probabilità di E_i , basterà conoscere la "storia" che lo precede.

→ principalmente si utilizzano modelli dei primi tre ordini

modello 0
presuppone l'indipendenza

$$P(A_1, \dots, A_N) = P(A_1) * \dots * P(N)$$

modello 1
presuppone la dipendenza dal primo elemento che prec.

$$P(A_1, \dots, A_N) = P(A_1) * P(A_2 | A_1) * \dots * P(N | N-1)$$

modello 2

presuppone la dipendenza dal bigramma precedente

$$P(A_1, \dots, A_N) = P(A_1) * P(A_2 | A_1) * \dots * P(A_3 | A_1, A_2) * \dots * P(A_N | A_{N-2}, A_{N-1})$$

ordine max
equivarrebbe alla chain rule
"useless"

→ le catene di Markov permettono di creare modelli probabilistici di sequenze linguistiche per le quali esistono particolari tipi di dipendenza

→ Anche per le catene di Markov è applicabile l'Add-one smoothing

$$P(B|A) = \frac{f(A, B) + 1}{f(A) + |V|}$$

→ Utilizzando un training corpus addestrato con catene di Markov, possiamo migliorare la generazione di un testo aumentando le dipendenze ad ogni ordine.

→ Utili per molte competenze del NLP come l'Automatic Speech Recognition, il Machine Translation e soprattutto la word prediction, simile all'MLE

Combinazioni

Alcune parole sono legate ad altre tramite forti lessicalizzazioni morfologiche e semantiche. Esse possono essere sostituite con altre parole per ottenere altre frasi grammaticali.

Combinazioni di due o più parole caratterizzate da un elevato grado di associazione, determinata dalla tendenza di "co-occorrere"

- argomenti o modificatori tipici
- argomenti o modificatori "idiosincratici"
- costruzioni idiomatiche
- nomi propri composti

Sono parole con un alto grado di associazione reciproca. Le misure di associazione lessicale:

quantificano la forza del legame tra due o più parole sul piano sintagmatico

La notzione intuitiva di associaz. lessicale viene trasformata in un indice quantitativo e misurabile

Esistono altri tipi di ~~combinazioni~~ combinazioni che si basano su legami non riconducibili a classi linguistiche generali. Sono difficili da sostituire perché produrebbero strani risultati.

Collocationi

tratti peculiari

Distinguibili in due categorie

empirico, o senso ampio, combinazioni ricorrenti e predibibili di parole osservate nell'uso linguistico (corpora)

teorico, o senso stretto, espressioni poliemetiche fortemente lessicalizzate, idiomatiche e idiosincratiche (multiword expressions)

→ elevata convenzionalità: sono tendenzialmente espressioni di usi convenzionali, tipici di varietà linguistiche

→ ridotta composizionalità semantica: non immediatamente ricavabile dalla composizione delle parole che lo formano
 $\{topolino grigio\} = \{\text{topolino}\} + \{\text{grigio}\}$
 $\{\text{gatta morta}\} \neq \{\text{gatta}\} + \{\text{morta}\}$

→ forte rigidità strutturale: sono spesso resistenti a modificazioni aggettivali o avverbiali, oppure occorrono solo in particolari forme flesse e contesti sintattici.

Collocazioni

- 1) Analisi linguistica del corpus: il testo deve essere almeno tokenizzato e possibilmente annotato con PoS tagging, lemmatizzazione, ecc.
 - 2) Selezione dei bigrammi: il tipo dei bigrammi che vengono selezionati dipende dal livello di annotazione.
 - 3) Costruzione della tabella di contingenza: dal totale dei bigrammi costruisco la tabella.
 - 4) Applicazione di una misura di associazione
 - 5) Ordinamento delle coppie in base alla forza di associazione
- Due parole si dicono fortemente associate quando si presentano più spesso insieme rispetto alle singole freq.

È necessario confrontare la frequenza osservata di $\langle u, v \rangle$ con la sua frequenza attesa (expected frequency), ossia la frequenza che ci aspettiamo qualora gli elementi fossero statisticamente indipendenti.

Un'altra versione con cui si calcola la MI è in termini di prob:
 $MI_{\langle x,y \rangle} = \log_2 \frac{p(x,y)}{p(x)p(y)} = \log_2 \frac{f(x,y)}{\frac{f(x)}{N} \frac{f(y)}{N}} = \frac{f(x,y) \cdot N}{f(x)f(y)}$

Estremamente sensibile agli eventi rari. In qualsiasi corpus bigrammi formati da rares avranno valori massimi

Se x e y ricorrono sempre assieme:

$$MI_{\langle x,y \rangle} = \frac{f \cdot N}{f^2} = \frac{N}{f}$$

Preso il bigramma $\langle x, y \rangle$ avremo

	X	\bar{x}	
Y	O_{11}	O_{12}	R_1
\bar{y}	O_{21}	O_{22}	R_2
	C_1	C_2	O_{22}

O_{11} : occorrono x e y
 O_{12} : occorre y , ma non x
 O_{21} : O_{22} occorre x , ma non y
 O_{22} : non occorre nessuno
 $N = R_1 + R_2 = C_1 + C_2$

Partendo dalla tabella di contingenza

	X	\bar{x}	
Y	E_{11}	E_{12}	
\bar{y}	E_{21}	E_{22}	

$E_{11} = (R_1, C_1)/N$
 $E_{12} = (R_1, C_2)/N$
 $E_{21} = (R_2, C_1)/N$
 $E_{22} = (R_2, C_2)/N$

Con entrambe le tabelle, possiamo calcolare la Mutual Information (MI) come

$$MI_{\langle x,y \rangle} = \log_2 \frac{O_{\langle x,y \rangle}}{E_{\langle x,y \rangle}} \quad \text{Se } MI \leq 0, \text{ c'è}$$

assenza di associazione tra le parole, altrimenti può essere forte.

La Local Mutual Information privilegia i bigrammi più frequenti ed è il termine fondamentale nel calcolo del Bayes-Ngram Ratio

$$LM_{\langle x,y \rangle} = f_{\langle x,y \rangle} - MI_{\langle x,y \rangle}$$

Legge di Bayes

Viene usata per decidere qual è l'ipotesi, o classe, più probabile I all'interno di \mathcal{I} che spiega un certo tipo di osservazioni O

$$\text{argmax}_{I \in \mathcal{I}} P(I|O) = \text{argmax}_{I \in \mathcal{I}} \frac{P(I) * P(O|I)}{P(O)}$$

Poiché stiamo cercando l'ipotesi più probabile data la stessa osservazione O , possiamo

$$\text{ignorare } P(O): \text{argmax}_{I \in \mathcal{I}} P(I) * P(O|I)$$

Ci permette di tradurre $P(I|O)$ in un prodotto di probabilità più facile da stimare

Chiamato teorema della probabilità delle cause, viene impiegato per calcolare la probabilità di una causa che ha provocato l'evento verificato.

Deriva dalla definizione di probabilità condizionata

$$P(A|B) = \frac{P(A \cap B)}{P(B)} \Rightarrow P(A \cap B) = P(B) * P(B|A)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)P(A)}{P(B)} \quad \text{dove:}$$

- $P(A)$: probabilità a priori
- $P(B|A)$: likelihood
- $P(A|B)$: probabilità a posteriori

Molti fenomeni linguistici possono essere modellati come processo di inferenza bayesiana, come:

Riconoscimento del parlato come l'Automatic Speech Recognition (ASR)

Problema di decidere a quale classe appartiene una certa osservazione linguistica

Noisy Channel Model

Un sistema dove si introduce un input in un canale "rumoroso", alterandone l'output. Il canale rumoroso contiene gli elementi dell'inferenza bayesiana, che produce in output una ipotesi con una certa probabilità.

Mentre le probabilità misurano fenomeni e sistemi il cui esito è incerto, l'entropia è una misura dell'incertezza di un fenomeno e misura la difficoltà nel predire l'esito.

Entropia

→ L'incertezza dipende da:

- numero di esiti alternativi possibili
- distribuzione delle probabilità per ciascun esito: in caso di esiti con probabilità uniforme è più difficile prevedere quello giusto.

→ L'informazione è la diminuzione dell'incertezza: se un evento aleatorio si verifica, avremo informazioni sia sugli esiti positivi che negativi.

→ L'entropia è la misura della quantità d'informazione portata dagli eventi prodotti da eventi aleatori, formalizzata come una variabile casuale.

Una Random Variable rappresenta un processo aleatorio, che è descritto dall'insieme di stati che esso può assumere e da una distribuzione di probabilità associata.

→ Secondo Shannon-Fano dice che: gli stati più probabili e frequenti sono descritti usando messaggi più corti, gli stati meno probabili usano messaggi più lunghi.

→ L'entropia puntuale, o informazione, di una parola corrisponde al numero di bit necessari per trasmettere che è stata estratta. Si misura come: $H(v) = -\log_2 p(v)^*$

→ Se un sistema W ha m stati equiprobabili, il numero di bit necessari per codificare ogni ~~messaggio~~ stato è $\log_2 m$.

Se gli stati sono equiprobabili vale che $p(v) = 1/m$ ed $m = 1/p(v)$. Per le proprietà logaritmiche:

$$H(v) = \log_2 1/p(v) = -\log_2 p(v)^*$$

→ Nel caso di stati non equiprobabili, l'entropia di un sistema W è definita come:

$$H(w) = -\sum p(v_i) \log_2 p(v_i)$$

→ Date le distribuzioni in due stati:

$$H(w) = -\sum p(w,t_i) \log_2 p(w,t_i)$$

Quella minore è più caratterizzante.

→ L'entropia aumenta con l'aumentare degli stati possibili di un sistema. A parità di stati possibili, l'entropia diminuisce se aumenta la "struttura" e l'"organizzazione" del sistema aumenta la predicitività delle dinamiche del sistema.

L'entropia per sequenze di parole:

$$H(w_i^*) = -\sum p(w_i^*) \log_2 p(w_i^*)$$

L'entropia per parola, o entropy rate:

$$H_{\text{rate}}(w_i^*) = \frac{1}{n} H(w_i^*)$$

Entropia

Shannon - McMillan - Breiman: se un linguaggio L è generato da un processo stocastico ergodico e stationario, allora vale che:

$$H_{\text{true}}(L) = \lim_{n \rightarrow \infty} -\frac{1}{n} \log_2 p(w_1, \dots, w_n)$$

Cross Entropy: $H(w_1^N, m) = -\sum p(w_1^N) \log_2 p(w_1^N)$

w_1^N è una var. casuale con distribuzione reale p
 m è un modello stocastico di w_1^N che cerca di approssimare la sua distribuzione reale.

Ci consente di misurare quanto bene un modello probabilistico approssima un certo processo stocastico:

$$H(w) \leq H(w, m)$$

È il costo di bits di usare m come modello per descrivere un processo con distribuzione p :

Per Shannon - McMillan - Breiman, possiamo approssimare la Cross Entropy prendendo un campione sufficientemente grande di testi del linguaggio come unica seq. di parole

$$H(p, m) = \lim_{n \rightarrow \infty} -\frac{1}{n} \log_2 m(w_1, \dots, w_n)$$

Il modello più accurato sarà quello con la cross entropy min.

È stationario se la probabilità che assegna a sequenze di parole sono invarianti rispetto al tempo

È ergodico se aumentando la lunghezza della sequenza di parole generate, possiamo ottenere un campione perfettamente rappresentativo del processo

In realtà il linguaggio non è né stationario né ergodico.

Natural language Processing

È un sistema in grado di accedere al contenuto di informazioni attraverso l'elaborazione del linguaggio.

Vengono effettuati dei task tipici della preparazione per l'analisi statistica

Diversi i problemi che possono presentarsi nell'addestramento degli algoritmi per via delle ambiguità fonologiche, morf., ecc.

Il machine learning comprende gli algoritmi che permettono alla macchina di imparare a svolgere un compito X , partendo da degli esempi su come svolgere quel determinato compito.

Machine learning

Usa modelli statisticci da dati nel corpus per costruire un modello per anticipare il testo. I componenti sono:

- training corpus ↗
- metodologia ↙
- testing corpus ↙

Se aggiunto al NLP, migliora di molto il suo funzionamento, ma rimane legato ai dati di partenza.
Esistono due tipi di algoritmi:

Supervisionato: esse vere basate su un tagging annotato e mano-tramite XML o simili. Adattati per le classificazioni, come trovare le concrete classi di importanza (machine readable)

Non supervisionato:
Si usano dei corpus non annotati, per creare modelli. Usati per campioni come il ranking dei dati in base a qualche funzione o il clustering in base a similitudine (raw corpora)

L'annotazione può essere fatta manuale, ma è molto lenta e costosa e a volte incisiva, o semi-automatica, con precisioni 95% o maggiori.

Processo di sviluppo di annotazione:
- Model, descrizione del fenomeno finito.
- Annotate, annotazione in base features
- Train, addestramento dell'algoritmo
- Test, test dell'algoritmo
- Evaluate, valutazione dell'esito
- Revise, revisione eventuale dell'algoritmo

Possono essere annotati morfo-sintatticamente o PoS tagging.

(analitica)
Altra annotazione tramite TreeBanks, che danno:
- bracketing, struttura di costituenti
- relazioni grammaticali
- struttura predicativa

Le scienze di annotazione:
- (labelled bracketing
[S [NP [Name] ...]]
- dependency structure

(semantica)
Named Entity Recognition è l'annotazione basata sulle categorie semantiche in quale assegna un nome proprio le "entità" della loro categoria (Person, Organ, ...)

BIO2: usata per espansioni poliemantiche
- B- begin
- I- la parte interna
- O- other

Annotatione

- I dati annotati "a mano" devono essere valutati per determinare il grado di affidabilità. Per essere tale, l'annotatione deve essere replicata in maniera coerente da più annotatori, anche in momenti diversi.

Il grado di accordo si basa sui dati di MINIMO due annotatori, interannotator agreement, ed esistono diverse misure. Nel caso di due annotatori, troviamo il K di Cohen. Più annotatori useranno multi-K, e altri.

- POS tagging usato per la dissibiguitazione multiorientistica. Il compito è quello di assegnare ad ogni token la categoria grammaticale, detta più complica nella specifica dei tratti morfologici.

- Poi usare la regola Pattern-action:
 - `(action)` seleziona e/o rimuove i tag
 - Il blocca `(action)` fino alla conferma
 - `<pattern>` se verificato, esegue `(action)`

- Nella fase di addestramento hanno training set, corpus di esempi annotati.
- feature: caratteristiche estratte dall'input
- lang model: `<feature, peso>`

→ K di Cohen misura l'accordo, al netto dell'accordo che si verificherebbe per caso. Per fare ciò, si crea la tabella di confusione e dove, nella diagonale, troveremo le volte che A1 e A2 sono concordi

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

		X	Y	
		X	50	20
		Y	10	20
C ₁	C ₂			
R ₁				
R ₂				

• $P(A)$ è il totale delle volte che concordano, diviso il totale degli elenchi

$$P(A) = (O_{11} + O_{22}) / N$$

• $P(c)$ è il totale delle volte che fanno le scelte per caso: $P(c) = P\{X\} + P\{Y\} \Rightarrow P\{X\} = R_1 R_2 * P(C_1)$, $P\{Y\} = R_2 R_1 * P(C_2)$

Per una buona accordanza, $K \geq 0,80$, e $0 \leq K \leq 1$. Se $K < 0,8$, il sistema andrebbe revisionato.

→ Negli algo supervisionati, il numero di classi y di un input x ($(x; y)$, coppia del training set) deve essere finito, mentre gli input quasi mai sono finiti. L'algo deve quindi imparare le generalizzazioni di eventi mai osservati.

Le features sono le caratteristiche che devono essere analizzate e sono indicate come coppia `(attr., value)`, dove il valore è un numero binario 0 o 1. L'estrazione delle features è uno dei compiti più difficili, poiché potrebbe falsificare il modello che viene creato.

Si distinguono in locali, estratte dal token stesso (forma, lessico, ecc), contestuali, estratte dal contesto in cui si trova il token (analizza la sua storia) e globali, più ampio di quelle contestuali ma meno usate (dominio del documento).

Gold Standard

→ È una porzione del test corpus annotato manualmente che rappresenta l'output di riferimento

→ La N-fold crossvalidation è la metodologia di valutazione ideale per quando i dati annotati sono limitati:

1. Si divide il corpus in N parti.
2. Si compiono N cicli di training-valutazione usando a ciclo una parte come test e le $n-1$ come training.
3. Si effettua la media delle prestazioni.

→ La baseline è il limite inferiore che ci si attende dal sistema. L'accuracy è invece data dal rapporto degli output corretti e la lunghezza del corpus

→ Information Retrieval: ci dà delle misure riguardanti:
 - precisione, la correttezza delle risposte del sistema
 - richiamo, la copertura del sistema

Natural Language Parsing

→ È il processo di analisi di una frase per determinare la sua struttura sintattica.
 Possono essere a costituenti o a dipendenze

de azioni effettuate per costruire

- shift: non c'è relazione tra i due tok, va avanti
- right: c'è relazione, il nodo sin. dipende da nd
- left: opposto di right

→ L'algoritmo termina dopo aver analizzato tutto e comincia l'analisi in base alle features estratte

→ Basandosi sugli output attestati nel GS:

- TP sono gli output confermati con il GS
- FP sono gli output errati nel GS
- FN sono gli output mancati ed errati ($FN = |N| - TP$)

$$\text{Precision} = \frac{TP}{TP+FP}$$

$$\text{Recall} = \frac{TP}{TP+FN}$$

Si penalizzano a vicenda: cresce una, diminuisce l'altra

F-Measure è la media armonica di P e R

$$F-M = \frac{2PR}{P+R}$$

→ Vengono individuate le relazioni sintattiche tra i token della frase. Producono alberi a dipendenze dove i nodi sono i token e la root, mentre gli archi le relazioni fra le parole.

→ Si usano classificatori addestrati su delle TreeBanks allo scopo di predire l'azione in base alle features.

Semantic Network

WordNet è un SN di tipo relazionale. Si può usare come dizionario, accedendo per lemma, o tesoro, accedendo per synset il quale ha un ID univoco.

Rappresenta il lessico computaz di maggior successo, con validatione su ampia scala di pregi e difetti delle reti semantiche.

Esistono dei cloni come:
- MultiWordNet
- ItalWordNet

da similità semantica si basa sulla distanza tra concetti, in base a #nodi/archi.

Sviluppato a Princeton, come modello relazionale riguardo nomi, agg. o verbi.

È una memoria semantica organizzata come una rete di nodi (concetti) e archi (relazioni). Si basano su una struttura tassonomica.

Un concetto viene rappresentato come l'insieme delle parole sinonime che lo esprimono, chiamato synset.

Un synset a cui appartiene il concetto X, può contenere una o più parole che lo esprimono. Una parola che può esprimere più concetti, genera ambiguità lessicale. Quest'ultima apparterrà ad n synset diversi: es. "cane"

FrameNet è un SN di tipo frame. Il significato di un'entrata lessicale viene descritto tramite un semantic frame

Rappresentazione schematica di un evento in termini dei possibili partecipanti (frame elements)
- ognuno associato alle sue possibili realizzazioni sintattiche
- per ogni realizzazione è associata una frase del corpus annotata

I lemmi ambigui sono associati a più frame

Gli elementi del frame element sono:
- lexical Unit coppia <parola, significato>
- Significato frame evocato da parola
- Frame, schema concettuale che descrive una scena coni sua FE:
• FE, i partecipanti, core FE - peripheral FE

Word Sense Disambiguation

Ha l'obiettivo di assegnare a ogni token il significato più appropriato in un contesto (es: BORSA)

Il funzionamento avviene con il principio Bayesiano su un vettore \vec{f} formato da al più 3 parole a sin e 3 a destra del token. Effettuando Bayes, si sceglie l'argmax prodotto