Laurea Specialistica in Informatica
a.a. 2006-2007

Interazione Uomo-Macchina II:

Interfacce Intelligenti

Nicole Novielli e Fiorella de Rosis

novielli_at_di.uniba.it

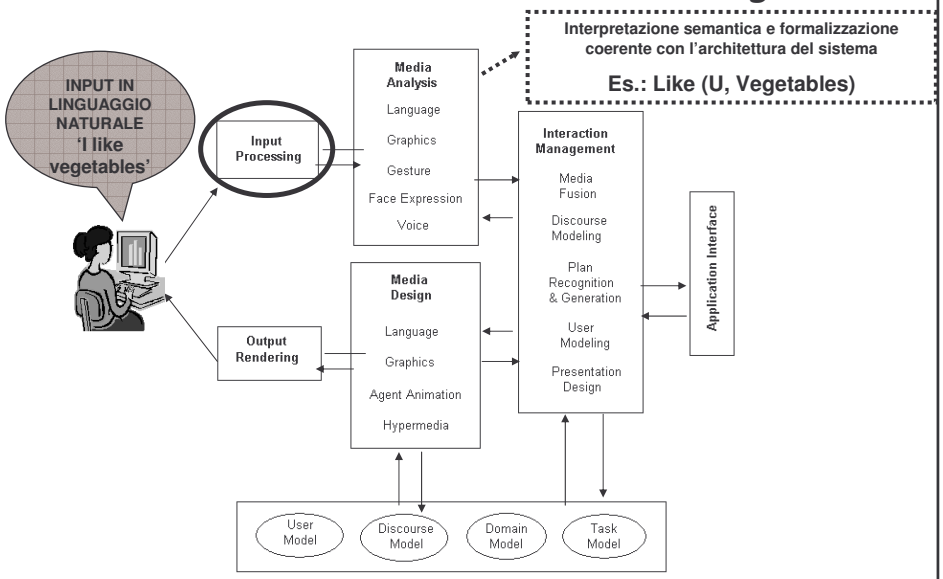
Introduzione
Prima parte: *Formalizzazione e Ragionamento*
- 1.1. Ragionamento logico:
  - Formalizzazione
  - Risoluzione
- 1.2. Ragionamento incerto
  - Reti Causali Probabilistiche
  - Reti dinamiche
  - Apprendimento di Reti

Seconda parte: *Modelli di Utente*
- 2.1. Modelli logici
- 2.2. Modelli con incertezza

Terza parte: *Interazione in linguaggio naturale*
- 3.1. Generazione di messaggi
  - Introduzione
  - Teorie
  - Metodi
- ***3.2. Comprensione di messaggi***

Quarta parte: *Simulazione di dialoghi*

Architettura di una interfaccia intelligente



Semantica

L'**obiettivo** del riconoscimento in un sistema di dialogo è quello di **formalizzare le frasi in linguaggio naturale tramite un linguaggio ad hoc** (ad es. linguaggio logico), allo scopo di 'ragionare' su di esse.

The dog ate.

Ate(dog)

A man saw a cat with a telescope

Saw(man, CatWithTelescope) gatto dotato di telescopio ☺

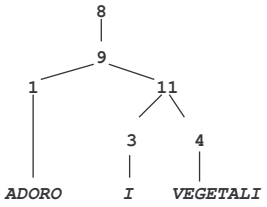
Oppure: Saw(man, Cat, UsingTelescope) usando un telescopio

nelle due interpretazioni della frase che abbiamo visto.

Jack downloaded the computer game

Downoaded(Jack, ComputerGame)

Nella lezione precedente abbiamo visto come realizzare il riconoscimento tramite PCFG (Probabilistic Context Free Grammar)



Output del parsing

```
(8 (9 (1 ADORO) (11 (3 I) (4 VEGETALI))))
```

Utilizzando la Symbol Table

Claim U, Like (U, Veg)

Formalizzazione logica

MA…

Problema

Per estrarre la semantica delle frasi dell'utente tramite grammatiche (probabilistiche), dovrei formalizzare **TUTTE** le possibili frasi del linguaggio naturale scelto per l'interazione

Il riconoscimento tramite PCFG pone delle forti limitazioni al modo di esprimersi dell'utente

Abbiamo bisogno di un metodo che ci consenta di svincolarci dalla rigida struttura sintattica delle frasi

Latent Semantic Analysis[1]

LSA (anche conosciuta come LSI, Latent Semantic Indexing) è una tecnica statistica molto conosciuta ed utilizzata nel campo del Natural Language Processing.

E' stata inizialmente sviluppata per valutare la similarita tra i documenti o tra i termini di un corpus in linguaggio naturale

Oggi è un metodo ampiamente utilizzato per l'information retrieval nelle collezioni di documenti in linguaggio naturale

Applicazioni: information retrieval, data clustering, text categorization, synonymy and polysemy

[1] Landauer, T.K. and Dumais, S.T.: A solution to Plato's problem: The Latent Semantic Analysis theory of the acquisition, induction, and representation of knowledge

Latent Semantic Analysis

Una possibile applicazione: i motori di ricerca su web (es. Altavista, Google)

Problema: Dato un insieme di documenti *D* ed una query *Q*, definire una funzione r_Q :D → **R** che associ, ad ogni pagina, un numero reale (rank), che indica il grado di *rilevanza* di quella pagina a fronte di quella query.

a. Fase di apprendimento: INDICIZZAZIONE della collezione di documenti. L'indicizzazione deve essere tale da consentire il ranking dei documenti in funzione della query utente!!!

b. Fase di utilizzo: inserimento di una query da parte dell'utente, ranking dei documenti in funzione della similarità della query utente con i vari documenti nella collezione

L'LSA utilizza una rappresentazione vettoriale dei documenti in linguaggio naturale

Indicizzazione dei documenti del corpus

$D = \{d_1, d_2, \ldots d_m\}$ insieme dei documenti da indicizzare

$T = \{t_1, t_2, \ldots t_n\}$ lessico considerato (es. tutti i termini del dizionario italiano, tutti I termini che compaiono in D, categorie semantiche come ad es. i 'sense' WordNet)

Costruisco la matrice termini x documenti *A*

$$a_{ij} = \begin{cases} 1 \text{ if } \text{contains}(d_j, t_i) = true \\ 0, otherwise \end{cases}$$

In realtà potrei usare indici più raffinati ad esempio frequenza del termine nel documento o tf-idf (li vedremo in seguito)

Terms	d1	d2	d3
a	1	1	1
arrived	0	1	1
damaged	1	0	0
delivery	0	1	0
fire	1	0	0
gold	1	0	1
in	1	1	1
of	1	1	1
shipment	1	0	1
silver	0	2	0
truck	0	1	1

A =

Rappresentazione dei documenti basata su Bag of Word (BoW): l'ordine in cui i termini $t \in T$ appaiono nel documento non è importante

Vector Space Model (VSM)

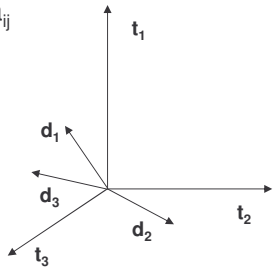
Il Vector Space Model ci consente di esprimere in maniera geometrica la rappresentazione del corpus basata su BoW

$D = \{d_1, d_2, \ldots d_m\}$

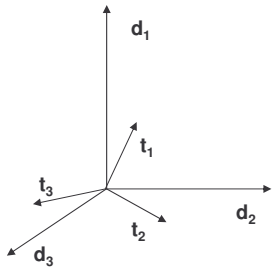
$T = \{t_1, t_2, \ldots t_n\}$ ⟹ Matrice termini x documenti *A*, di dimensioni *n x m,* con a_{ij} frequenza di t_i in d_j

Il VSM è uno spazio *n*-dimensionale $\Re^n$ in cui il documento d_j è rappresentato tramite il vettore $\vec{a}_j$ tale che la sua *i*-esima componente coincide con a_{ij}

(1)



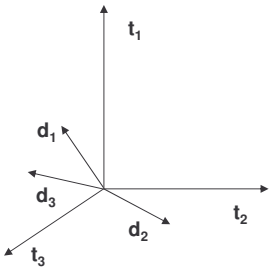
(2)



Nota: il VSM per la rappresentazione dei documenti in funzione dei termini (1) e il VSM per la rappresentazione dei termini in funzione dei documenti (destra) sono **spazi vettoriali disgiunti**!! (scelgo quale usare in funzione del task)

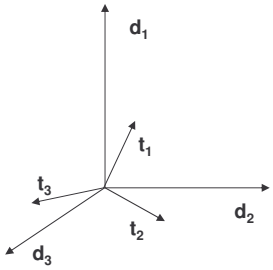
Scelgo quale usare VSM (dei documenti o dei termini) in funzione del task da realizzare, ad esempio:

(1)



Task di information retrieval, document categorization (etc.): agisco sui documenti

(2)



Studio della sinonimia o della polisemia, definizione di domini semantici (etc.): agisco sul lessico

Come calcolare la similarità

Terms	d1	d2	d3
a	1	1	1
arrived	0	1	1
damaged	1	0	0
delivery	0	1	0
fire	1	0	0
gold	1	0	1
in	1	1	1
of	1	1	1
shipment	1	0	1
silver	0	2	0
truck	0	1	1

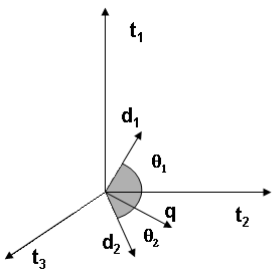
A =

q
0
0
0
0
0
1
0
0
0
1
1

q =

Formalizzo la query allo stesso modo…

… e calcolo per ogni documento la sua similarità con la query calcolando il coseno dell'angolo formato dai due vettori



$$simil(\vec{d}_1, \vec{q}) = \cos\theta_1 = \frac{\vec{d}_1 \, x \, \vec{q}}{\left\|\vec{d}_1\right\| \left\|\vec{q}\right\|}$$

il valore massimo di similarità è 1 (cos 0°=1) (vettori coincidenti)
Per vettori semanticamente indipendenti (ortogonali) il coseno si avvicina a 0 (cos 90° = 0)

Cos θ_2 > Cos θ_1 quindi il documento d_2 è più rilevante del documento d_1 rispetto alla query utente q (ranking dei documenti in output)

ALCUNI PROBLEMI

In entrambi i casi (rappresentazione dei documenti in funzione dei termini o, viceversa, dei termini in funzioni dei documenti) la rapprsentazione tramite VSM è affetta da alcuni problemi.

a. In molti casi, è stato dimostrato che **non necessariamente un dizionario dei termini T più ricco** corrisponde, a parità di condizioni, ad un **incremento delle performance**

b. Non tutti i termini possiedono lo stesso potenziale descrittivo rispetto ad un determinato dominio di applicazione o argomento.

E' il caso di parole molto frequenti, come ad esempio *congiunzioni, verbi ausiliari,* gran parte degli *avverbi* (le cosiddette '***stop-word***') E' molto frequente perciò il ricorso a procedure di 'stop-word elimination', ossia procedure di '***cleaning***' del corpus di documenti preliminari alla creazione della matrice termini x documenti

Vengono spesso eliminati quei ***termini che ricorrono raramente*** nel corpus (*hapax legomena*, alla lettera 'pronunciate una volta sola'). Potrebbe trattarsi di ***errori ortografici*** o di ***neologismi*** che non sono ancora entrati nel vocabolario comune e che pertanto non ha senso includere nel dizionario dei termini T

E' fondamentale definire il dizionario dei termini in modo ragionato e coerente con il task

Definizione del dizionario dei termini (lessico)

La definizione del lessico T da utilizzare nella costruzione della matrice è un problema cruciale

A seconda del task di riconoscimento che desidero attuare posso:

• costruire un **lessico 'grezzo'** senza introdurre nessun elemento di conoscenza sul dominio, al più estraendo la radice delle parole (stemming)

Es.: "biostatistics" diventa "biostat"

• decidere di eliminare tutte quelle parole che in alcuni domini non sono considerate rilevanti (**stopword elimination**). Tipica procedura di preprocessing del dataset nei task di information retrieval

Esempi di stopword list http://members.unine.ch/jacques.savoy/clef/index.html

• aggiungere informazioni sul **ruolo sintattico** del termine nell'ambito della frase

Es.: termini arricchiti con informazioni sulla POS

In questo modo t_1 (voto, first-person verb) è diverso da t_2 (voto, noun)

Il VSM non è in grado di risolvere i problemi legati all'ambiguità ed alla polisemia del linguaggio naturale

ES.: le due frasi "*He is affected by AIDS*" and "*HIV is a virus*" non hanno nessuna parola in comune

Nel VSM la loro similarità è pari a zero perché le due frasi sono rappresentate da vettori ortogonali (cos 90° = 0), nonostante i concetti espressi siano strettamente correlati

Al contrario, trovo similarità tra le due frasi "*The laptop has been infected by a virus*" and "*HIV is a virus*" a causa dell'ambiguità della parola 'virus' che nei due contesti assume significati differenti

Per risolvere (o almeno arginare) questo tipo di confondimenti potrei definire il lessico **considerando la semantica** delle espressioni (parole e/o insiemi di parole) in linguaggio naturale nella definizione del lessico

Potrei:

• decidere di creare un **dizionario semantico dei termini** per la collezione di documenti: i termini in T corrispondono alla rappresentazione semantica della parola (ad es. i 'sense' di WordNet)

Gli elementi di T saranno gli ID dei 'sense' nel dizionario semantico di riferimento: ad ogni singolo ID sono associati uno o più sinonimi

• definire delle **categorie semantiche ad hoc**, in accordo con l'obiettivo da raggiungere

ad esempio, volendo riconoscere i turni di dialogo in cui l'utente si presenta al sistema potremmo definire la categoria Self-Introduction, in cui includiamo gruppi di *n* parole che rappresentano tutte le formule di presentazione che l'utente può usare

Self-Introduction = {mi_chiamo, il_mio_nome_è, io_sono … }

EGreeting = {buongiorno, buondì … }

EAgName = {Valentina, cara … }

Indicizzazione: valutare la co-occorrenza

In precedenza abbiamo detto che nella matrice A, termini x documenti

$$a_{ij} = \begin{cases} 1 \text{ if } \text{contains}(d_j, t_i) = true \\ otherwise \end{cases}$$

In realtà si utilizzano misure più raffinate

• tf_{ij} *(term-Frequency)*, ossia la frequenza del termine *i*-esimo nel documento *j*-esimo ($freq_{ij}$) rispetto alla cardinalità *m* del corpus

• *tf-idf (Frequency/Inverse Document Frequency* by Salton*, 1989)* calcolato come

$$tf - idf_{ij} = tf_{ij} * idf_{ij} = \frac{freq_{ij}}{l} * (\log \frac{m}{r})$$

cardinalità della collezione

numero di item in cui il termine compare

Osservazione

Il tf/idf normalizza la Term Frequency rispetto all'Inverse Document Frequency, ossia in funzione del numero di documenti in cui il termine appare, a livello dell'intera collezione.

CHE SIGNIFICA? $tf / idf_{ij} = tf_{ij} * (\log \frac{N}{n})$

Es. vogliamo calcolare il tf-idf del termine *h-esimo* per il documento *k-esimo*. Nel corpus osservo che t_h compare in TUTTI i documenti della collezione. Quindi $idf_{hk} = \log 1 = 0$, quindi anche $tf\text{-}idf_{hk}$

-> UN TERMINE CHE APPARE IN TUTTI I DOCUMENTI DELLA COLLEZIONE NON è SIGNIFICATIVAMENTE RILEVANTE PER NESSUNO DI ESSI, IN PARTICOLARE

Indicizzazione: definizione dell'unità di testo

Non sempre l'unità di testo analizzata coincide con un documento

In accordo con le necessità legate al task ed allo scenario posso decidere che gli elementi di D, ossia i vari dn corrispondono a

- intere pagine web
- frammenti di lunghezza prestabilita di documenti
- turni di dialogo (mosse), se lo scenario di riferimento è quello di un'interfaccia intelligente che simula un dialogo tra utente e sistema

Nella parte relativa all' esercitazione 2b vedremo come l'LSA può essere impiegata per estrarre la semantica delle mosse di dialogo in linguaggio naturale eseguite degli utenti

Sparsità della matrice

La principale limitazione della rappresentazione nel VSM è la *sparsità*.

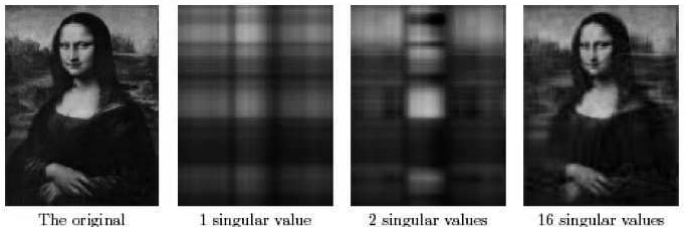
corrispondenza		segni						
segno/categoria	categorie	fsi	ffwell	qagt	talks	pcom	ncom	fstyle
fsi, ffwell	Ciao = {Ciao, cià,…}	0,82	0,52	0,05	0,03	0,07	0,00	0,06
fsi	Egreeting = {buongiorno, buondì,…}	0,09	0,00	0,00	0,01	0,00	0,00	0,01
	ESelfIntrod = {mi chiamo, io sono, …}	0,45	0,00	0,00	0,01	0,00	0,00	0,01
ffwell	EBye = {arrivederci, a presto, …}	0,00	0,35	0,03	0,00	0,00	0,03	0,03
	EThanks = {grazie, ti ringrazio, …}	0,00	0,39	0,05	0,01	0,19	0,00	0,08
negcom	Objections = {ma, però, …}	0,00	0,04	0,14	0,12	0,04	0,31	0,13
	NegEvalOfAgent = {non sai, non capisci,…}	0,00	0,00	0,08	0,00	0,00	0,08	0,02
poscom	EAgreement = {concordo, …}	0,00	0,00	0,12	0,04	0,41	0,08	0,10
	EAttitude = {amo, mi piace, …}	0,00	0,00	0,00	0,00	0,11	0,00	0,02
	…	…	…	…	…	…	…	…

Nella matrice appaiono molti valori nulli, nonostante la definizione di un lessico basato su categorie semantiche e la bassa dimensionalità

Singular Value Decomposition

La principale limitazione della rappresentazione nel VSM è la *sparsità.*

Un modo per risolvere le limitazioni nelle performance che derivano da questo problema è applicare la Singular Value Decomposition (SVD) alla matrice termini x documenti A, allo scopo di ottenerne una rappresentazione in uno spazio vettoriale di minori dimensioni (riduzione del rango della matrice termini x documenti)



Singular Value Decomposition per la riduzione di un'immagine bitmap[1]

[1] Alfio Gliozzo, 2005 - Semantic Domains in Computational Linguistics, PhD thesis, University of Trento

L'SVD scompone una matrice (nel nostro caso la matrice termini x documenti A) nel prodotto di tre nuove matrici

$$(1) \quad A = U\Sigma V^T$$

dove U e V^T sono matrici ortogonali[1] di dimensioni, rispettivamente (*m x m*) ed (*n x n*); le colonne di U rappresentano gli autovalori di A^TA e le colonne di V^T rappresentano gli autovettori di AA^T.

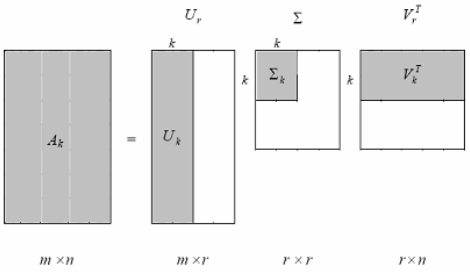
$$\Sigma = \begin{vmatrix} D & 0 \\ 0 & 0 \end{vmatrix}$$ in cui D = diag $(\sigma_1, \sigma_2, \ldots \sigma_r)$ con $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_r \geq 0$

r = rango di A — *Valori singolari di A*

Si parla di valori singolari e non di autovalori poiché il procedimento deve essere generalizzato e applicabile anche a matrici rettangolari

A questo punto scelgo un intero k $\leq$ *r* = rank(A) e costruisco la matrice A_k

$$(2) \quad A_k = \sum_{i=1}^{k} \sigma_i u_i v_i^t$$



[1] In matematica una matrice ortogonale è una matrice quadrata G la cui trasposta coincide con la sua inversa ossia $G^*G^t=G^{t*}G = I_n$

Cosa fa quindi l'SVD?

Prende in input una matrice rettangolare A e la scompone nel modo che segue

(1) $A = U\Sigma V^T$

Applicando la (2) **approssima** A ad una matrice A_k di rango *k* inferiore

(2) $\mathrm{A}_k = \sum_{i=1}^{k} \sigma_i u_i v_i^t$

Il fattore *k*, rango della matrice output dell'SVD, rappresenta la percentuale di informazione della matrice A che ho mantenunto nell'approssimarla ad A_k

L'SVD consente di ottenere, data una matrice A di rango *r*, la sua migliore approssimazione A_k, di rango $k<r$

Largamente impiegata nei problemi di compressione di dati (eliminazione di 'rumore' nei dati, compressione di immagini etc…)

Analisi del confondimento

problema legati all'ambiguità ed alla polisemia del linguaggio naturale

ES.: riconoscimento di segni di social attitude in move utente in linguaggio naturale

Segni di social attitude: Friendly Self Introduction, Friendly Farewell, Positive and Negative Comment, Colloquial Style, Talks About Self, Question to the Agent

Sulla base dell'annotazione di tre rater indipendenti, etichetto un database di move utente (dialoghi di Wizard of OZ), in cui ogni turno di dialogo dell'utente riceve una o più label di social attitude, secondo il criterio del majority voting

Definisco le categorie linguistiche rilevanti per ognuno dei segni secondo il concetto di 'salience': se $\text{freq}(\text{categoria}_i|\text{segno}_h) > \text{freq}(\text{categoria}_i)$ allora la categoria i-esima è 'salient' per il segno j-esimo

Costruisco la matrice termini x documenti, utilizzando come dizionario dei termini l'insieme delle categorie lessicali e come unità di testo l'insieme delle mosse etichettate con il medesimo segno di social attitude

→

corrispondenza		segni						
segno/categoria	categorie	fsi	ffwell	qagt	talks	pcom	ncom	fstyle
fsi, ffwell	Ciao = {Ciao, cià,...}	0,82	0,52	0,05	0,03	0,07	0,00	0,06
fsi	Egreeting = {buongiorno, buondì,...}	0,09	0,00	0,00	0,01	0,00	0,00	0,01
	ESelfIntrod = {mi chiamo, io sono, ...}	0,45	0,00	0,00	0,01	0,00	0,00	0,01
ffwell	EBye = {arrivederci, a presto, ...}	0,00	0,35	0,03	0,00	0,00	0,03	0,03
	EThanks = {grazie, ti ringrazio, ...}	0,00	0,39	0,05	0,01	0,19	0,00	0,08
negcom	Objections = {ma, però, ...}	0,00	0,04	0,14	0,12	0,04	0,31	0,13
	NegEvalOfAgent = {non sai, non capisci,...}	0,00	0,00	0,08	0,00	0,00	0,08	0,02
poscom	EAgreement = {concordo, ...}	0,00	0,00	0,12	0,04	0,41	0,08	0,10
	EAttitude = {amo, mi piace, ...}	0,00	0,00	0,00	0,00	0,11	0,00	0,02
	...	...	...	...	...	...	...	...

Possibili cause di confondimento - 1:

***Overlapping parziale* tra il contenuto di alcune categorie semantiche.**
Es.: La categoria '*Ciao*', è rilevante (tf-idf elevato) per 'Friendly Self Introduction' e 'Friendly Farewell'. 'Mio*'* è nella categoria 'Self-introduction' e 'First-person pronouns'. Perciò, *'Ciao, il mio nome è Carlo'* è riconosciuta come simile, nello stesso tempo, a 'Friendly self introduction', 'Friendly farewell' e 'Talks about self'.

Altre possibili cause di confondimento

Overlapping parziale tra alcune sequenze di parole in alcune categorie semantiche:

2: Ad es.: e.g., '*sei'* è nella categoria 'Second-person auxiliary verbs' mentre 'sei maleducata*'* è nella categoria che racchiude le espressioni di 'Evaluation of agent's politeness'.
'Sei maleducata' sarà quindi classificata come *Negative comment* ma anche come *question to the agent*.
Un problema simile si riscontra nel confondimento tra 'Friendly farewell' e 'Question about the agent' per la move '*ciao, ti ringrazio per avermi dedicato un pò del tuo tempo virtuale*' che include 'Ciao' e 'Second-person pronoun'.

aspecificity of some components of semantic categories:

3:
i punti esclamativi sono parte della categoria *'paralanguage'* ma non si trovano soltanto nelle move di Colloquial Style (*'greet!'*), ma anche nel Friendly Farewell *('See you soon then!'),* nei Positive comment *('I agree!'),* ad in move etichettate con label relative ad altri segni di social attitude