POLITECNICO DI MILANO

Scuola di Ingegneria Industriale e dell'Informazione
Corso di Laurea Magistrale in INGEGNERIA MATEMATICA

Nuove strategie per la Sentiment Analysis applicata ai Social Network

Analisi dei tweet su Expo Milano 2015

Relatore:
Prof. Simone Vantini

Tesi di Laurea Magistrale di:
Mattia Satta
matr. 799563

Anno Accademico 2014-2015

"YOU ARE WHAT YOU TWEET!"
- Germany Kent

"Grazie al social networking, anche la reazione di un singolo consumatore a un prodotto si trasforma in una forza che potrebbe innescare un boicottaggio oppure avviare affari d'oro per nuove imprese."
- Daniel Goleman, Intelligenza ecologica, 2009

"Abbiamo trasformato lentamente i social network, senza che molti media se ne accorgessero troppo, da luoghi dell'esibizionismo, luoghi del mostrarsi per quello che si fa, in luoghi dell'anima, in luoghi dove finalmente l'identità prende un suo ruolo sempre più importante, e dove il privato e il pubblico sono divisi sempre meno, sono una linea sottile, fragile, impercettibile."
- Roberto Cotroneo, su Sette, 2012

Abstract

L'obiettivo di questa tesi è quello di applicare la sentiment analysis ai social network, analizzando i metodi preesistenti e formulando delle nuove strategie per determinare un classificatore più preciso ed accurato.
Dopo un breve excursus sulla letteratura si presenterà il metodo di classificazione aggregata di Hopkins e King, che risulta essere quello più adatto all'obiettivo prefissato, valutandone tutte le sue caratteristiche, i vantaggi e gli svantaggi. Successivamente si propongono alcune strategie innovative che da una parte sfruttano la classificazione classica, detta individuale, per selezionare e pesare opportunamente le parole dei testi, dall'altra si propongono di ridurre la complessità testuale tramite la sostituzione dei termini con i sinonimi, con l'obiettivo, in entrambe i casi, di migliorare le prestazioni della classificazione.

Si valutano quindi i metodi applicandoli allo studio del sentiment espresso dalla popolazione Twitter sull'esposizione universale *Expo Milano 2015, Nutrire il Pianeta, Energia per la Vita*. I risultati evidenziano che le nuove strategie portano in alcuni casi a dei miglioramenti rispetto alle strategie preesistenti, arrivando quindi alla conclusione che la strada percorsa è buona, ma deve essere approfondita e valutata meglio.

In conclusione sono stati discussi tutti i problemi che ci sono stati e che devono essere ancora risolti, riguardanti principalmente l'aspetto pratico di trattare con variabili rappresentate da stringhe di caratteri e quindi i possibili sviluppi futuri.

Indice

Introduzione

Da sempre si è cercato di trovare dei metodi validi per classificare i testi, ma la vera sentiment analysis nasce circa 15 anni fa con l'avvento delle pagine web. Il concetto di sentiment è utilizzato poiché lo scopo principale è quello di capire il sentimento o il parere delle persone, analizzando i testi che essi scrivono, con strumenti matematici e statistici. Gli esempi più semplici e comunemente usati sono sentiment *positivo, negativo o neutro*, ma si potrebbe parlare anche di *arrabbiato, soddisfatto, in disaccordo, frustrato, indifferente* e tanti altri. L'oggetto di studio è appunto il testo, quindi le singole parole che lo compongono e il modo tramite il quale sono legate tra loro.
È ovvio che leggendo il testo si può capire se esprime un parere e di che natura è, ma se si pensa ad un testo molto lungo oppure a un'infinità di testi allora si capisce che sono necessari degli strumenti automatici che siano in grado di fare queste analisi. Inizialmente la sentiment analysis venne usata per analizzare i pareri che le persone esprimevano sui blog: Turney (2002) fa classificazione supervisionata sulla base delle recensioni di automobili, film, viaggi e banche, classificando quindi l'oggetto in questione come raccomandato o non raccomandato; anche in Pang and Lee (2005) si ha lo stesso obiettivo, usando come scala di riferimento del gradimento le stelle (da 0 a 5, ancora presenti negli app-stores per valutare il gradimento delle applicazioni che gli utenti scaricano).

Ultimamente la sentiment analysis è diventata molto importante vista la massiccia diffusione ed importanza acquistata dai social network quali Twitter, Facebook e Google+. Nel caso di Twitter mensilmente 288 milioni di utenti sono attivi e 500 milioni sono i tweet che ogni giorno sono inviati (about.twitter.com). Questi social network sono diventati un mezzo tramite il quale gli utenti si possono esprimere, non solo si riportano i momenti positivi e negativi passati durante la giornata, ma si esprimono pareri e giudizi su eventuali prodotti commerciali provati, programmi tv o radio seguiti e si commentano le notizie che arrivano da tutto il mondo. Di conseguenza per un'azienda che crea un certo prodotto o una società che fornisce un certo servizio diventa fondamentale usare le informazioni che arrivano dalla rete per capire il parere dei consumatori (o almeno una parte di essi). Sempre recentemente la sentiment analysis è spesso usata in ambito politico, ad

esempio per prevedere gli esiti delle votazioni oppure testare il gradimento dei cittadini su un certo tema.
Certamente sono sempre esistiti strumenti come i sondaggi telefonici, il servizio ai clienti e altri canali che mettono in contatto diretto produttore e consumatore, ma i vantaggi dell'analisi dei social sono differenti, tra i quali: costo pressoché nullo, rapidità nell'ottenere i dati, pareri spontanei e sinceri, campione molto più ampio e composto da persone di ogni tipo. Ovviamente ha anche i suoi limiti, per esempio la maggior parte delle volte non si conosce la natura del campione analizzato, quindi non si hanno informazioni sull'età, sesso, grado di istruzione degli utenti e tante informazioni che potrebbero essere utili.
Quindi si inizia a capire che trovare metodi e strumenti per la sentiment analysis, che siano sempre più precisi ed accurati, è molto importante.

I metodi usati per la sentiment analysis, in quanto metodi di classificazione, posso essere usati appunto per fare pura classificazione di testi, anche in categorie del tutto generiche. Ovviamente ogni metodo ha un campo di applicabilità in cui le performance sono migliori, bisogna quindi trovare quel metodo che meglio si adatta al tipo di problema che si sta affrontando e al tipo di dati che si hanno a disposizione.

Il lavoro che segue si propone di presentare lo stato dell'arte esistente per la sentiment analysis sui social network e cercare di migliorarlo proponendo alcune strategie innovative, con lo scopo di migliorare l'accuratezza del classificatore. In particolare la tesi è strutturata nel seguente modo: all'inizio si fa una breve descrizione dei principali metodi esistenti per fare sentiment analysis e si descrivono gli approcci che si possono utilizzare (cap.1). Nel capitolo 2 si andrà ad approfondire il metodo di Hopkins and King (2010), mentre nel capitolo 3 si propongono le strategie innovative. La prima propone di usare la tecnica di classificazione degli alberi (Random Forest in particolare), la seconda invece la sostituzione delle parole tramite i sinonimi. Per finire nel capitolo 4 si applicano le tecniche studiate per cercare di analizzare il sentiment legato all'Expo 2015 di Milano e per valutarne gli aspetti critici che possono essere migliorati. Si conclude quindi con tutti i commenti finali, le problematiche avute e gli sviluppi futuri.

Capitolo 1

Letteratura

Come detto questo tipo di analisi sono nate solo recentemente. Gli approcci usati sono svariati: si va da metodi si *scoring* a metodi di analisi testuale, da metodi in cui le categorie sono note e fissate a priori ad altri in cui le categorie sono incognite. Si farà un riassunto espositivo delle tecniche più note, ordinando l'esposizione in modo logico ma non temporale, come presentato da Ceron et al. (2014). Si precisa che di questi metodi esistono diverse varianti, qui si presentano i casi più semplici per fare un quadro generale.

1.1 Scoring

In questi metodi il testo non è visto come un discorso che deve essere interpretato e capito, ma semplicemente come un insieme di parole che danno un'informazione su dove si trova il testo in un certo spazio dimensionale, ad ognuno viene assegnato un punteggio (da cui il nome *scoring*), quindi si dispongono tali testi su una retta in base al punteggio. Lo scopo è quello di creare un ordinamento fra i testi che poi, in base al problema, deve essere opportunamente interpretato. Un caso semplice potrebbe essere quello di suddividere i testi in positivi, negativi e neutri se i punteggi assegnati sono rispettivamente molto maggiore di 0, molto minore di 0 e prossimi allo 0 (vedi la figura 1.1). Di tali metodi esistono sia metodi supervisionati, sia metodi non supervisionati.

Wordscores. Un metodo supervisionato è l'approccio *Wordscores* presentato da Laver et al. (2002), spesso usato per prevedere le posizioni politiche dei partiti. Basandosi sui conteggi delle parole e sui punteggi noti di alcuni documenti, si assegnano i punteggi a tutti gli altri documenti senza fare particolari assunzioni distribuzionali o funzionali.

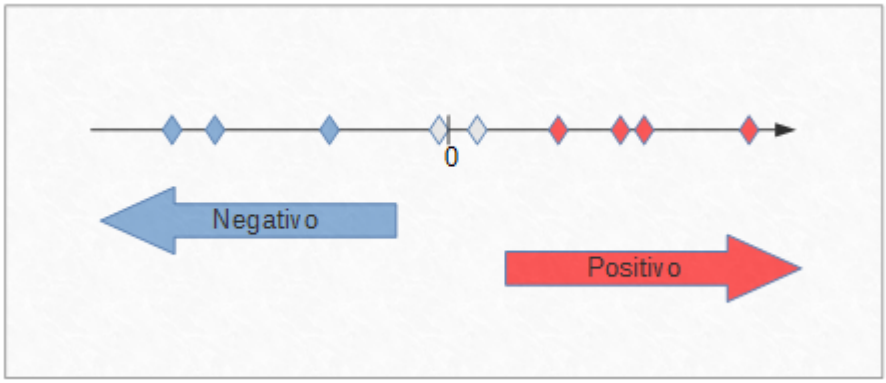


Figura 1.1: *Scoring: i testi sono disposti sulla retta in base al punteggio assegnato. Semplice esempio in cui si assegna la categoria positivo ai testi che stanno sulla destra, negativo a quelli sulla sinistra, nella zona centrale invece giacciono i testi che vengono considerati neutri.*

Wordfish. Uno dei metodi non supervisionati è basato sull'algoritmo chiamato *Wordfish* e presentato da Slapin and Proksch (2008). Anche questo metodo è stato creato per prevedere o analizzare la serie storica delle posizioni assunte dai diversi partiti ed è basato sulla frequenza delle singole parole. A differenza del caso precedente, si assume che i conteggi delle parole nei testi abbiano una certa distribuzione, per cui non si necessita di nessun training set. Per discriminare i testi appartenenti alle varie categorie deve essere fatta un'analisi a posteriori leggendo il loro contenuto, infatti essendo un metodo non supervisionate non erano note in principio.

1.2 Analisi Testuale

Questi metodi si basano sull'analisi del testo, per cui il contenuto delle parole diventa importante, non solo la presenza o assenza delle stesse o il numero di volte che compare nel testo. Anche in questo caso si ha una distinzione tra il caso in cui le categorie sono note ed assegnate in partenza e il caso in cui sono del tutto incognite e verranno create solo a posteriori sulla base dei risultati ottenuti.

1.2.1 Dizionari ontologici

Questo è un metodo di analisi testuale in cui le categorie sono note e fissate a priori ed è forse il più semplice e comunemente utilizzato, fornisce un tipo di analisi base, anche se in certi contesti molto efficace. Ancor prima di leggere ed analizzare i testi si procede stilando un elenco delle parole associate a ciascuna categoria di sentiment, a questo punto leggendo una frase si procede col contare il numero di parole appartenenti a ciascuna categoria e poi si utilizzano delle regole decisionali per capire a quale classe assegnare il testo. Un esempio potrebbe essere il caso semplice di positivo, negativo e neutro in cui si conta un +1 per ogni parola positiva, un -1 per ogni parola negativa. Si procede con la somma e se il risultato è positivo si assegna la categoria *positivo*, viceversa *negativo*, mentre si assegna la classe neutrale se il risultato è nullo.
Quindi si può dire che è un metodo poco sofisticato, che può essere usato per analisi semplici e basilari in cui si vuole avere un'idea di massima sul sentiment espresso. Si noti inoltre che è necessario avere a disposizione un cospicuo numero di parole per formare l'elenco delle parole che esprimono i diversi sentiment.

Esempi:

- *"Questo film è bello, ma ripetitivo!"*
  Alla parola *bello* si assegna +1, mentre alla parola *ripetitivo* -1; in totale si ottiene 0, quindi il testo viene considerato neutro.

- *"Questa macchina non solo è brutta, ma è pure scomoda!"*
  Alla parola *brutta* e *scomoda* vengono assegnati due -1 per un totale di -2, di conseguenza il testo è considerato negativo.

1.2.2 LDA: Latent Dirichlet Allocation

Questo metodo fa parte di una classe più ampia di metodi detti *Topic Models* (Blei et al., 2003)), in cui le categorie (in questo caso più comunemente chiamate topics o argomenti) sono incognite in tutto il dataset e si assume che ciascun testo sia una mistura di vari argomenti. Seguendo l'approccio Bayesiano, ciascun argomento ha una propria distribuzione a priori di Dirichlet e ad ognuno di essi è assegnato un gruppo di parole (con distribuzione multinomiale). Ciascun testo è quindi ottenuto scegliendo prima di tutto un topic e solo successivamente le relative parole in base al topic scelto. Con le tecniche Bayesiane si stimano le distribuzioni a posteriori sia dei topic sia delle parole, dopo di che, con un processo inverso, sulla base delle parole si assegnano le probabilità che ciascun documento sia attribuito a ciascun argomento.

1.2.3 Metodi supervisionati

Questi metodi, in quanto supervisionati, sono tali che si costruirà un classificatore tramite la classificazione, manuale o automatica, di un gruppo di testi. In base al tipo di approccio che si vuole avere, e in base a quello che è l'utilizzo che si vuol fare della sentiment analysis, si procede alla suddivisione dei metodi in due classi che prendono il nome di *individuale* ed *aggregata*. I nomi derivano dal fatto che nella classificazione individuale si procede ad assegnare la categoria ad ogni singolo testo dell'intero corpus di testi che si ha a disposizione, mentre nella classificazione aggregata si ottiene un informazione più compatta e riassuntiva che consiste nell'intera distribuzione di probabilità di testi in ciascuna categoria di sentiment.
È chiaro quindi che tutto dipende dallo scopo dello studio. Se si è interessati a sapere a quale classe appartiene un certo testo o un gruppo di testi si procede con quella individuale, mentre se si sta studiando un grande numero di testi e si vuole capire quanti testi vengono classificati come appartenenti ad una certa classe, allora si procede con quella aggregata. Si può facilmente passare da classificazione individuale ad aggregata con un processo di tipo *CC* (*Classify and Count*), come mostrato nella figura 1.2, ma il discorso è più delicato di quel che si può pensare, infatti facendo una classificazione individuale e poi contando il numero di testi assegnati a ciascuna categoria si amplificano gli errori di misclassificazione rispetto alla classificazione individuale. Si affronterà questo aspetto nel capitolo 2.

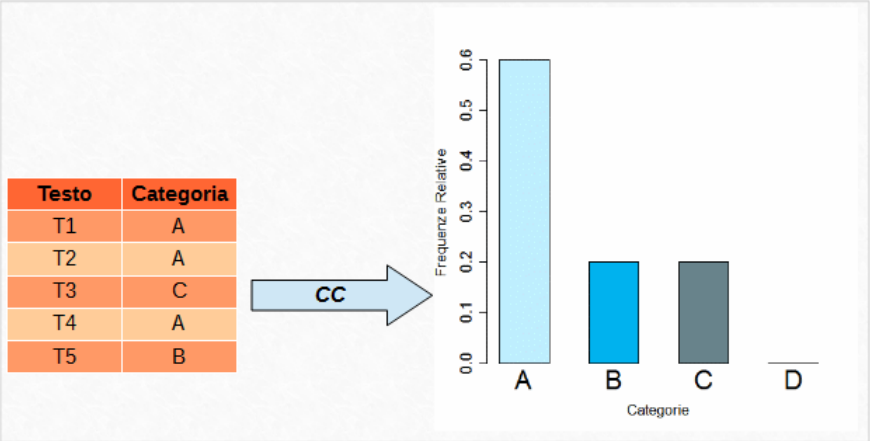


Figura 1.2: *Classify and Count: dalla classificazione individuale si passa alla classificazione aggregata*

Metodi di classificazione individuale. I metodi di classificazione individuale (i principali sono elencati e spiegati in Pang et al. (2002)) sono principalmente metodi di machine learning, quindi richiedono che parte dei testi che costituiranno il training set siano già etichettati. Il metodo più sicuro ma dispendioso è il tagging manuale, per rendere il processo più veloce spesso si utilizza un tagging automatico: in Go et al. (2009) per esempio si assegnano le categorie in base alle emotions presenti nel testo, altri invece usano il metodo con i dizionari ontologici descritto poco fa.
I principali metodi individuali sono:

- CART e Random Forest: classificano i testi usando delle regole dicotomiche, costruendo un vero e proprio albero delle decisioni (Breiman et al., 1984).

- Naive Bayes: usa il teorema di Bayes per valutare la probabilità che un certo documento appartenga a ciascuna delle categorie, assegnando successivamente la categoria con probabilità maggiore. Il punto centrale è l'assunzione che ogni parola del testo sia indipendente dalle altre (cosa però poco realistica).

- Maximum Entropy: è un metodo che, come suggerisce il nome, massimizza l'entropia di una distribuzione rispettando dei vincoli derivanti da quantità calcolate con il training set. Non facendo nessuna assunzione sull'indipendenza tra le parole, spesso è più accurato rispetto al precedente.

- Support Vector Machine (SVM): è sicuramente il metodo individuale più usato poiché il più efficiente nella classificazione tradizionale di testi. In questo metodo si cerca di trovare il miglior iperpiano separatore (o più iperpiani) tra le nuvole di punti, massimizzando quanto più possibile la distanza.

Metodi di classificazione aggregata. I metodi di classificazione aggregata sono nati solo recentemente e diventati fondamentali proprio grazie all'avvento dei social network. Questi metodi si basano sull'approccio descritto da Hopkins and King (2010). La grande novità è che l'errore di misclassificazione decresce rispetto all'aggregazione dei risultati ottenuti da classificazione individuale, ottenendo così un metodo molto più preciso ed anche più veloce visto che non c'è il passaggio intermedio della classificazione individuale. Poiché è un metodo supervisionato sarà sempre necessario fare una classificazione manuale di alcuni testi e questo sarà il vero e proprio punto debole del metodo.

L'analisi che si vuole fare è basata proprio sulla classificazione aggregata.

Nei capitoli successivi descriveremo con precisione sia l'aggregazione dei risultati della classificazione individuale e i problemi legati al *CC*, sia il metodo di Hopkins e King e presenteremo due aspetti innovativi volti a migliorare l'accuratezza e precisione dello stesso metodo.

1.3 Preprocessamento del testo

Si presentano ora alcune tecniche di preprocessamento del testo che vengono comunemente utilizzate prima dell'applicazione dei metodi. Non è un passo obbligatorio, anzi in alcune tecniche non è necessario, ma è fortemente consigliato in tante altre.
In questo tipo di studi è ovvio che le variabili di interesse siano le parole, la cui combinazione determina il testo. Affinché queste si possano usare in tutti i metodi statistici e non, devono essere tramutate in variabili numeriche. Ma poiché il linguaggio, qualsiasi sia la lingua, è molto complesso e articolato, il testo deve essere prima *depurato* da tutto ciò che non è utile per questo tipo di studi. Si procede allora con un preprocessamento del testo che, ovviamente, porterà a perdita di informazioni. Infatti, sia per la lingua italiana che per le altre lingue, è noto che l'ordine delle parole e la presenza di una preposizione piuttosto che un'altra alteri completamente il significato di una frase, ma affinché si possano usare dei metodi automatici bisogna semplificare la complessa struttura del linguaggio. Per tali motivi si deve essere molto cauti e precisi nell'eseguire tale procedimento, specificando ogni singolo passo, poiché le scelte fatte in partenza andranno ad influenzare i risultati finali.

Si riportano di seguito il tipo di manipolazioni che si possono eseguire e la relativa motivazione, segue l'esempio nella tabella 1.1

- all'interno di una frase sono presenti una serie di parole che hanno un ruolo differente, ci si deve concentrare su quelle che ci aiutano a capire il sentiment del testo, mentre devono essere eliminate da tutto il corpus quelle che non portano informazioni importanti. Queste parole si definiscono *stopwords* ed è l'insieme delle preposizioni personali, preposizioni semplici e composte, gli articoli, i verbi ausiliari, ecc. (vedi le appendici C e D). Stabilire quali siano esattamente le stopwords non è facile e dipende dal contesto in cui si sta lavorando. La scelta inoltre dipende dalla lingua, come si può osservare dalle appendici C e D in italiano ci sono molte più stopwords rispetto all'inglese.
  Questo elenco è nato per ottimizzare il lavoro dei motori di ricerca che, ignorando le stopwords, risultano più rapidi. Ovviamente ogni motore di ricerca avrà un elenco differente o comunque userà queste parole in modo differente.

- stessa cosa per la punteggiatura: sebbene sia fondamentale per capire il significato del testo, quando si utilizzano dei metodi matematico-statistici come quelli che vedremo è buona pratica eliminare tutta la punteggiatura (inclusi eventuali simboli come &, $ ecc.). Solo in alcuni casi possono essere tenuti particolari simboli se sono necessari nella classificazione del testo e questo dipende dalla natura dei testi analizzati e dal contesto nel quale si sta lavorando (per esempio si potrebbe essere interessati a conservare gli hashtag (#) dei tweet).

- tutte le lettere vengono rese minuscole: ciò che conta è la parola in se e non come questa viene scritta. Anzi per facilitare il riconoscimento delle parole è necessario che tutte le lettere siano rappresentate in modo unico.

- si rimuovono tutti gli spazi bianchi in eccesso tra le parole: questo è principalmente una buona abitudine, ma è anche necessario se si vuole lavorare con alcune funzioni che sono sensibili al carattere che separa ciascuna parola dalla successiva e/o dalla precedente.

- *stemming*: questa è una pratica che viene eseguita su ogni singola parola del testo. Per esprimere un concetto in ogni lingua ci sono tante parole che hanno la stessa radice, di conseguenza si punta l'attenzione sulla radice delle parole. Per chiarezza si consideri il seguente esempio: *famiglia, famigliare, famiglie* vengono sostituite dalla radice *famigl-*. Il concetto espresso è lo stesso, ma usando la radice si ha una notevole semplificazione (tre variabili distinte sono sostituite da un unico rappresentante). Lo stemmer comunemente utilizzato è quello di Porter, l'algoritmo che viene seguito per eseguire la procedura di stemming è descritta da van Rijsbergen et al. (1980).

Bisogna fare attenzione ad un aspetto che Pang et al. (2002) hanno descritto ed analizzato a fondo. Come detto la variabile di interesse è la parola e su di essa si eseguono tutte le possibili manipolazioni. Molto spesso però l'insieme di due parole acquisisce un senso ben più forte ed interessante rispetto al problema che si sta studiando. Per esempio consideriamo le parole *bianca* e *casa*, se nel testo vengono trovate come *"casa ... bianca"* il significato più semplice è che si sta descrivendo o indicando una casa che è bianca, se invece si trova *casa bianca* in sequenza allora è abbastanza chiaro che si stia parlando della residenza ufficiale del presidente degli Stati Uniti a Washington. Per questo si introducono gli unigram (*casa*, *bianca*, qualsiasi altra parola presa singolarmente) e anche i bigram, trigram ecc. (*casa bianca*, *studio ovale*, ecc.).

Esempio: *#polimi una delle migliori università d'Italia, inoltre un neolaureato trova subito lavoro!*

Azione	**Risultato**
Tutte le lettere minuscole	#polimi una delle migliori università d'italia, inoltre un neolaureato trova subito lavoro!
Rimozione stopwords	#polimi ___ _____ migliori università _'italia, inoltre __ neolaureato trova subito lavoro!
Rimozione punteggiatura	_polimi ___ _____ migliori università __italia_ inoltre __ neolaureato trova subito lavoro_
Stemming	_polim ___ _____ miglior univers __ital_ inoltr __ neolaur trov sub lavor_
Rimozione degli spazi bianchi in eccesso	polim miglior univers ital inoltr neolaur trov sub lavor

Tabella 1.1: *Sequenza del preprocessamento del testo nell'esempio.*

Pang et al. (2002) dimostrano comunque che l'utilizzo di bigram, trigram o n-gram in generale, unito all'uso di unigram, non porta un grande miglioramento del metodo, ovvero gli unigram da soli riescono a dare tutte le informazioni necessarie. Nella maggior parte dei casi si procederà quindi concentrandosi solo sugli unigram, che verranno considerati come singole parole.

Questi passaggi appena descritti hanno contribuito alla semplificazione del testo che comunque non è ancora pronto ad assere trattato matematicamente. Per trasformare ciascuna frase in una variabile numerica, una volta fatto lo stemming, si considerano tutti gli stem trovati nel corpus, dopo di che si procede nel seguente modo: per ogni testo si va a vedere quale stem è presente e quale no costruendo un vettore di 0 (assenza dello stem) o 1 (presenza dello stem). La sequenza di 0/1 relativa a ciascun testo è detta *word stem profile* (si veda l'esempio nella tabella 1.2).
Questo lavoro lo si fa per ogni testo del corpus: se si assegna ciascun testo alle righe di una matrice e ciascuno stem alle colonne, tutta l'informazione del corpus viene racchiusa all'interno di tale matrice. A partire dalla matrice infatti sarà sempre possibile ricostruire ciascun testo. È ovvio che in base al metodo utilizzato per analizzare i testi si userà tale matrice in modo differente.

Esempio:
Testo 1: *Mi piace la vostra nuova casa!*
Testo 2: *La casa è vecchia, ma la porta è nuova.*
Testo 3: *La Casa Bianca ha 132 stanze.*
Testo 4: *Non aprite quella porta!*

Testo \ Stem	piac-	nuov-	cas-	vecc-	port-	bianc-	stanz-	aprit-
1	1	1	1	0	0	0	0	0
2	0	1	1	1	1	0	0	0
3	0	0	1	0	0	1	1	0
4	0	0	0	0	1	0	0	1

Tabella 1.2: *Matrice di 0/1 per il corpus di testi nell'esempio.*

Capitolo 2

Dalla classificazione individuale al metodo di Hopkins e King

2.1 Notazione e introduzione delle variabili

Come già detto questo metodo è supervisionato, di conseguenza verrà utilizzato un training set (ed un testing set per valutare la bontà del classificatore), tutti i testi che fanno parte di questi insiemi sono stati etichettati manualmente.
Si indichi con i ciascun testo etichettato, per un totale di n testi, quindi $i = 1, ..., n$. Sia invece l ciascun testo del dataset, per un totale di m testi, quindi $l = 1, ..., m$. Come già spiegato in precedenza $m >> n$, il numero di testi etichettati sarà solo un piccola parte di tutti i testi che si hanno a disposizione. Si discuterà più avanti sulla scelta dei testi da etichettare. Sia ora $j = 1, ..., J$ l'indice delle categorie di testi, si introduce allora la prima variabile discreta D_i, quindi l'espressione $D_i = j$ dice che la categoria del testo i è la j-esima.
La seconda variabile che si introduce è $B_{i,k}$, questa è dicotomica a valori in {0,1} e assume valore 1 se lo stem k è presente nel testo i (con $k = 1, ..., K$), 0 altrimenti. Si osservi che l'insieme $\{B_{i,1}, ..., B_{i,K}\}$ descrive completamente un testo e non è nient'altro che lo *word stem profile* del documento i, che indicheremo tramite il vettore $\mathbf{S_i}$. In generale si indicherà con $\mathbf{S}$ il vettore di tutti gli word stem profile possibili (lungo quindi $2^K \times 1$).

Lo scopo del metodo sarà quello di ricavare la distribuzione di tutti i testi $\mathbf{P(D)} = \{\mathbf{P(D = 1)}, \mathbf{P(D = 2)}, ..., \mathbf{P(D = J)}\}$,ovvero la massa di probabilità assegnata all'insieme delle categorie.

2.2 Aggregazione

Partendo dai risultati della classificazione individuale, che consistono in un vettore che indica la classe assegnata a ciascun testo del dataset, la prima cosa che verrebbe da fare è quella di andare a vedere le proporzioni dei testi in ciascuna categoria. Quindi stimare $P(D = j)$ con:

$$\frac{1}{m}\sum_{l=1}^{m}\mathbb{1}(\hat{D}_l = j) \tag{2.1}$$

in cui si indica con $\hat{D}_l$ la categoria assegnata al testo l dal classificatore individuale. Questo modo di procedere, che è stato denominato *classify and count*, è sicuramente il più naturale, ma se lo scopo dello studio è ricavare la massa di probabilità, non è quello più adatto. Il problema principale risulta essere legato all'amplificazione dell'errore. Qualsiasi sia la procedura tramite il quale si effettua la classificazione individuale, si incorre comunque in un certo errore di misclassificazione, il problema è che aggregando questi risultati l'errore tende ad aumentare. Di conseguenza si può dire che il tutto non è per forza la somma delle parti. Si dovrebbe cercare quindi di fare una classificazione individuale con errori bassissimi.

Uno dei problemi per cui avviene questo può essere dovuto al modo tramite il quale si lavora. Come spiegato si parte dal vettore S degli stem profile per poi assegnare la classe D, quindi si lavora su $P(D|S)$, ma questo procedimento è opposto a quello che succede nella formulazione ed esplicazione di un sentimento o di un parere. Infatti prima nasce nelle persone il sentimento o il parere su una questione, dopo di che, tramite le parole e il loro uso, si cerca di esprimerli, quindi si dovrebbe lavorare su oggetto del tipo $P(S|D)$. Si noti anche che, affinché il metodo individuale funzioni, devono essere rispettate due assunzioni, la prima delle quali è che la classe di modelli scelta per $P(D|S)$ deve contenere quella *vera*.
La seconda assunzione è che l'insieme degli stem profile S deve ben rappresentare ciascuna categoria D (Hand, 2006), questa assunzione non risulta però mai verificata nel caso del linguaggio umano poiché S è per costruzione un sottoinsieme degli stem profile usati nel linguaggio. Comunque, anche assumendo che queste ipotesi valgano, si ha un problema di bias tra la proporzione vera e quella stimata di ciascuna categoria, che non è legata all'errore di misclassificazione. Si deve quindi cercare un modo differente di affrontare il problema per poter ottenere delle proporzioni il più veritiere possibile.

2.3 Aggregazione *corretta*

Hopkins e King propongono una correzione che consente di ricavare delle stime corrette delle proporzioni delle categorie a partire dai risultati della classificazione individuale. Il metodo lavora sui testi del testing set: $\hat{D}$ rappresenta la stima delle etichette fatta tramite il metodo individuale, mentre D le etichette assegnate agli stessi testi tramite hand coding. L'errore di misclassificazione sarà dato da $P(\hat{D}_i = j|D_i = j)$: lo scopo è quello di usare tale informazione per cercare di rendere corretta la stima complessiva. Si formalizza il discorso nel caso di due categorie, quindi $\hat{D}$ e D sono dicotomiche, si prendano a valori in $\{1, 2\}$. Si consideri ora $P(\hat{D} = 1|D = 1)$, comunemente chiamata *sensibilità* e $P(\hat{D} = 2|D = 2)$, comunemente chiamata *specificità*. Usando il teorema delle probabilità totali si avrà:

$$P(\hat{D} = 1) = (sens)P(D = 1) + (1 - spec)P(D = 2) \tag{2.2}$$

Poiché lo scopo è quello di ricavare la vera proporzione di testi che sta in ciascuna categoria e poiché $P(D = 1) = 1 - P(D = 2)$, invertendo la (2.2) si ricava:

$$P(D = 1) = \frac{P(\hat{D} = 1) - (1 - spec)}{sens - (1 - spec)} \tag{2.3}$$

Si valutano quindi sensibilità e specificità usando training e testing set, mentre si calcola $P(\hat{D} = 1)$ con il metodo individuale usando tutti i testi a disposizione e aggregando i risultati.

Questo discorso si può generalizzare al caso in cui ci siano più di due categorie, la (2.2) sarà pari al seguente sistema di equazioni per ogni j in $\{1, ..., J\}$:

$$P(\hat{D} = j) = \sum_{j'=1}^{J} P(\hat{D} = j|D = j')P(D = j') \tag{2.4}$$

In realtà si useranno $J - 1$ equazioni di tale sistema, l'ultima sarà data dal fatto che $P(D)$ deve sommare a 1, in quanto probabilità. Il vantaggio di lavorare in questo modo è che non è stata richiesta nessuna ipotesi aggiuntiva particolare, ne tanto meno si è richiesto di incrementare l'accuratezza dei metodi individuali, producendo comunque delle stime non distorte. L'unica cosa che si richiede è che le $P(\hat{D} = j|D = j')$, ricavate dalla cross-validazione, si mantengano uguali anche nell'insieme dei testi non etichettati. Questa ipotesi si considera comunque assunta poiché è una richiesta di tutti i metodi individuali, se quindi non fosse verificata anche il metodo individuale sarebbe scorretto.

2.4 Metodo di Hopkins e King

Si presenta ora il metodo non parametrico fornito da Hopkins and King (2010), senza passare per la classificazione individuale. Invece di valutare $\hat{D}$ tramite S e D (usando i metodi individuali) e apportare poi la correzione vista nel paragrafo precedente, nella formula (2.4) si utilizza direttamente S invece di $\hat{D}$. Questo è naturale poiché $\hat{D}$ è funzione di S, come già detto le parole vengono scelte per esprimere l'appartenenza del testo ad una certa categoria. Si arriva quindi alla seguente espressione:

$$P(S = s) = \sum_{j=1}^{J} P(S = s|D = j)P(D = j) \tag{2.5}$$

Riscrivendo la (2.5) sotto forma matriciale diventa:

$$\mathbf{P}(\mathbf{S}) = \mathbf{P}(\mathbf{S}|\mathbf{D})\mathbf{P}(\mathbf{D}) \tag{2.6}$$

Si ricordi che $\mathbf{S}$ è l'insieme degli word stem profiles, quindi $\mathbf{P}(\mathbf{S})$ ha dimensione $2^K \times 1$, $\mathbf{D}$ è il vettore delle categorie, quindi $\mathbf{P}(\mathbf{D})$, che è il vettore incognito, ha dimensione J$\times$1, mentre $\mathbf{P}(\mathbf{S}|\mathbf{D})$ è la matrice delle probabilità di trovare un certo profilo in una data categoria di testi, quindi ha dimensione $2^K \times J$.

Come detto la quantità incognita di interesse è $\mathbf{P(D)}$, mentre $\mathbf{P(S)}$ e $\mathbf{P(S|D)}$ devono essere stimate dai dati. $\mathbf{P(S)}$ si stima tranquillamente dal tutto il corpus di testi che si hanno a disposizione, andando quindi a valutare la frequenza di ciascun word stem profile nel dataset. Per $\mathbf{P(S|D)}$ la questione è più delicata, infatti per fare la sua stima si deve essere in possesso delle etichette, di conseguenza andrà fatta sui soli testi che sono stati etichettati a mano. Indicando con $\mathbf{P^h(S|D)}$ la stima empirica fatta sui testi hand coded, si fa l'assunzione che le due matrici di probabilità coincidano, quindi:

$$\mathbf{P^h}(\mathbf{S}|\mathbf{D}) = \mathbf{P}(\mathbf{S}|\mathbf{D}) \tag{2.7}$$

In questo modo si stima questa quantità valutando la frequenza di ciascun stem profile in ciascuna categoria, usando i testi del training set. Si osservi che questa assunzione è più debole rispetto a quelle richieste dalla classificazione individuale, ma soprattutto deve essere ben chiaro che non si richiede che la distribuzione degli stem profile nell'hand coded set sia pari a quella vera. Non deve esserci infatti un campione statisticamente rappresentativo, ma piuttosto linguisticamente rappresentativo. In ciascuna categoria dell'insieme dei testi etichettati manualmente deve esserci una variabilità nell'uso degli word stem profile tale da rispecchiare quelli che comunemente sono usati in tutto il dataset.

Osservando ora la (2.6), per ricavare $\mathbf{P}(\mathbf{D})$, ottenendo direttamente le proporzioni dei documenti in ciascuna categoria e onde evitare problemi di invertibilità della matrice $\mathbf{P}(\mathbf{S}|\mathbf{D})$, si inverte la formula nel seguente modo:

$$\mathbf{P}(\mathbf{D}) = (\mathbf{P}(\mathbf{S}|\mathbf{D})'\mathbf{P}(\mathbf{S}|\mathbf{D}))^{-1}\mathbf{P}(\mathbf{S}|\mathbf{D})'\mathbf{P}(\mathbf{S}) \tag{2.8}$$

2.4.1 Scelta delle categorie e tagging manuale

La scelta delle categorie a primo impatto sembra una cosa naturale, in base al problema che si vuole affrontare nascono le categorie. Ma la questione è molto più delicata: le categorie scelte infatti devono essere mutuamente esclusive, esaustive ed omogenee. Questo vuol dire che nel caso di sentiment positivo o negativo bisogna anche introdurre la categoria di testi neutri, ovvero quei testi che esprimono un'opinione ma questa è neutrale. Altra categoria interessante e spesso usata è quella dei testi che non esprimono un'opinione nonostante trattino temi relativi al problema che si sta studiando. In pratica in base al dataset che si ha a disposizione la cosa fondamentale è che, nel momento in cui si effettua il tagging manuale, si deve essere sempre in grado di etichettare ciascun testo, senza avere alcun dubbio.

La categoria *Off Topic*. In molte applicazioni è necessario introdurre un'ulteriore categoria che racchiuda tutti i testi che sono fuori tema rispetto al problema che si sta affrontando. Questi sono molto comuni nel caso in cui si stia lavorando su testi scaricati dai social network, in tali casi infatti la ricerca viene fatta tramite parole chiave che, anche se scelte in modo accurato, catturano qualsiasi testo le contenga, ma non sempre questi testi sono utili. Quindi per rispettare la regole citate sopra di dover essere sempre in grado di sistemare ciascun testo in una delle categorie, è bene introdurre questa categoria per questa tipologia di dati.

Tagging manuale. Il processo di assegnazione dei testi a ciascuna categoria è molto delicato e richiede la massima attenzione, da questo infatti dipendono poi i risultati finali. Si ricorda che il classificatore impara le regole decisionali che lo portano a stimare le percentuali richieste dal set di testi etichettati manualmente, quindi questa procedura deve essere fatta con la massima attenzione e cura. A tal proposito chi effettua l'hand coding deve avere ben presente quale sia il problema che si sta affrontando, quali sono le categorie che sono state scelte e non avere nessun dubbio sull'assegnazione da fare. L'ideale sarebbe quello di farsi consigliare o assegnare questo lavoro ad un esperto.

Nonostante questo possono esserci comunque due problemi, il primo dei quali è l'errore umano dovuto a distrazione, digitazione sbagliata o altri problemi analoghi. Il secondo è invece legato alla soggettività della scelta fatta, infatti per un individuo il testo i potrebbe appartenere alla categoria j mentre per un altro individuo lo stesso testo potrebbe appartenere alla categoria $\tilde{j}$. È allora necessario far eseguire l'etichettatura manuale a più persone, possibilmente in sedi separate. Queste stesse persone poi devono discutere i punti su cui si è in disaccordo e arrivare ad una versione comune, nel caso di indecisione è bene eliminare quel testo dall'hand coded set.
Per semplificare si potrebbe dire che se già le persone hanno dei dubbi su quale sia la classe da assegnare ad un testo, lo strumento statistico presentato non potrà certo fare di meglio, quindi non si può pretendere di ottenere un metodo accurato se già l'essere umano non procede con accuratezza. È molto importante quindi che il training e il testing set siano costituiti da testi affidabili, ovvero che contengono le informazioni necessarie a costruire e testare il classificatore (se ve ne sono alcuni che non portano nessuna informazione e non si è in grado di classificarli, devono essere eliminati dal dataset). Inoltre i testi devono essere ben codificati in modo da garantire una buona riuscita del metodo.
È facile capire che questa è la parte più dispendiosa, specialmente in tempo, di tutto il metodo.

2.4.2 Verifica delle ipotesi

L'unica assunzione fatta è che $\mathbf{P^h}(\mathbf{S}|\mathbf{D}) = \mathbf{P}(\mathbf{S}|\mathbf{D})$ (2.7), ovvero nel momento in cui si etichettano manualmente i testi del training set si deve essere certi che questi siano ben rappresentativi del linguaggio usato in tutto il corpus di testi. In pratica nel training set devono essere presenti un numero cospicuo e rappresentativo di stem profile tali da poter ben rappresentare e caratterizzare ciascuna delle categorie. Un caso in cui questa ipotesi potrebbe risultare non verificata è quello della *population drift*, che potrebbe esserci per esempio quando si crea il training set tanto tempo prima rispetto all'acquisizione dei testi del corpus che verranno poi utilizzati per fare la stima delle proporzioni. Quindi tutto dipende dal tipo di analisi che si sta eseguendo:

- analisi retrospettiva: si sta analizzando un insieme di testi che riguardano un problema del passato, allora siamo certi che prendendo casualmente dei testi da questo insieme l'ipotesi (2.7) sia verificata.

- analisi predittiva: si vuole utilizzare un insieme di testi per costruire un classificatore da poter utilizzare per predire eventi futuri, usando dei testi che verranno scritti successivamente. È il caso in cui si vogliano studiare le serie temporali, in cui gli avvenimenti si susseguono molto velocemente nel tempo, per esempio le elezioni politiche. In questo caso

si deve stare attenti al fatto che col passare del tempo non ci sia una population drift, in tal caso è buona norma continuare ad aggiornare il training set con testi che stiano "al passo" con il linguaggio usato dal resto del dataset.

Si osserva che non è necessario che in ciascuna categoria ci sia lo stesso numero di testi, ma si richiede che ciascuna di essa sia ben rappresentata, per cui è da sconsigliare un numero di testi eccessivamente basso. Se si ha a che fare con delle categorie che, per loro natura, sono poco numerose si procede alla ricerca di appositi testi da inserire nel training set in modo da arricchire la variabilità di testi nella categoria. Se non fosse possibile si può procedere cambiando alcune regole di catalogazione in modo da evitare la presenza di categorie poco numerose.

2.4.3 Problemi computazionali

Si noti che nella (2.8) si utilizza una matrice di dimensioni $2^K \times J$ ed in generale K è molto grande, di conseguenza si devono fare dei calcoli con matrici di grandi dimensioni che richiedono un ampio costo computazionale. Inoltre $n << 2^K$, ovvero il numero di testi del training set e quindi di stem profiles usati per stimare $\mathbf{P(S|D)}$, è molto minore rispetto al numero totale di stem profiles possibili sulla base dei K stem presenti nel testo, questo porta a delle probabilità molto piccole. Per evitare quindi costi computazionali eccessivamente grandi e il problema della sparsità si ricorre al metodo Monte Carlo proposto da King and Lu (2008) che consiste nel *bagging*: si applica il metodo finora descritto utilizzando dei sottoinsiemi di stem più piccoli formati da pochi elementi (si sceglie tra 5 a 25 stem ciascuno, dipende al dataset). Quindi si ricava per ognuno la stima di $\mathbf{P(D)}$, dopo di che si procede a fare una media aritmetica tra le quantità ottenute.

È ovvio che diventa molto importante il modo tramite il quale si sceglie il numero e quali stem inserire in ciascun sottoinsieme. Per quanto riguarda quest'ultimo aspetto Hopkins e King scelgono casualmente gli stem assegnando ad ognuno la stessa probabilità di essere pescato. In alcuni casi potrebbe essere utile assegnare a ciascuno stem un peso, per cui stem più *importanti* hanno maggiori probabilità di essere estratti e appartenere a vari sottoinsiemi. Un sistema di pesi usato dai due autori è inversamente proporzionale alla varianza della variabile aleatoria associata a ciascuno stem. Sia X_j la variabile aleatoria binaria associata allo stem j, che vale 1 se lo stem è presente e 0 altrimenti, allora essa ha distribuzione bernoulli di parametro q_j, quindi il peso sarà $p_j = \frac{\frac{1}{q_j(1-q_j)}}{\sum_{j=1}^{J} \frac{1}{q_j(1-q_j)}}$. La stima di q_j è fatta ovviamente in modo empirico sulla base dei testi che si hanno a disposizione. Si noti che in

questo modo si assegna un peso maggiore a stem che compaiono raramente oppure molto spesso, mentre un peso basso ai casi intermedi.
Per quanto riguarda la numerosità invece si procede con le classiche tecniche di cross-validazione, usando training e testing set per scegliere la numerosità ottimale dei sottoinsiemi che garantisca maggiore accuratezza del metodo.

2.4.4 Quanti testi etichettare

Un'altra questione interessante riguarda il numero di testi che devono essere etichettati manualmente. Come si è detto questo procedimento è il più dispendioso, quindi si deve cercare di avere un numero di testi non eccessivamente grande, ricordando però che gli stessi testi devono essere abbastanza da poter garantire l'ipotesi (2.7).

Hopkins e King eseguono un po' di prove e concludono che, analizzando i blogs che si occupano di politica e classificando i testi in nove categorie differenti, etichettare 100 testi porta ad un errore molto basso, che continua a diminuire se si classificano manualmente altri testi. In alcuni casi però si potrà aver bisogno di codificare un numero molto maggiore di testi, bisogna sempre controllare che l'assunzione (2.7) sia rispettata, per cui se alcune categorie non vengono descritte in modo adeguato si dovrà procedere a taggare altri testi fino a quando non si raggiunge un numero di testi e una variabilità linguistica sufficientemente adeguati in ciascuna categoria. Questo succede per esempio se si scaricano dei testi da piattaforme pubbliche come i social network, in cui si può trovare materiale di ogni genere, per cui per raggiungere la numerosità adeguata si dovranno leggere molti più testi di quelli precedentemente indicati.

2.4.5 Vantaggi e svantaggi del metodo

Se si rivede nell'insieme il metodo appena descritto ci si rende subito conto che questo è di facile comprensione e applicazione, se i testi fossero pochi saremmo in grado di applicare i passi descritti anche manualmente. Nonostante questo le potenzialità del metodo sono impressionanti. Nei metodi presentati in precedenza infatti l'analisi partiva quasi sempre con la scelta delle parole (o stem) da cercare nei testi del dataset, dopo di che si procedeva con la descrizione di un metodo che a partire dalle parole assegna una certa categoria. Il problema è che il gruppo di parole si sceglie a prescindere dai testi del corpus, spesso si aveva un approccio del tipo: *brutto, osceno, orribile* sono parole che esprimono un sentimento negativo, mentre *bello, felice, incantevole* sono parole che esprimono un sentimento positivo. L'aspetto positivo del Metodo di Hopkins e King è che in modo automatico riesce a catturare queste parole, ma anche tante altre che a priori non erano state

considerate.

Ma la vera forza del metodo è l'uso delle combinazioni delle parole: non si vanno ad utilizzare gli stem da soli, ma si verifica la presenza contemporanea di più stem nella frase, ovvero si conta il numero di volte che un certo word stem profile compare nel corpus. Quindi si punta l'attenzione sulla presenza combinata di differenti stem e ci si basa sul fatto che il sentiment non è espresso dalle singole parole, ma dalla loro combinazione. Questo permette, per esempio, di riuscire a riconoscere e classificare in modo giusto anche i testi ironici, oppure riconoscere particolari espressioni che potrebbero caratterizzare una certa categoria e che fanno parte del bagaglio di conoscenze di chi fa il tagging manuale e non dovuto all'appartenenza semantica delle varie parole ad una certa classe.

Lo svantaggio più grande è sicuramente il processo di hand coding già citato e discusso in precedenza, non solo in quanto processo costoso in termini di tempo, ma anche perché si deve essere in grado di accertare che la (2.7) sia verificata. Questa assunzione come detto risulta essere più debole rispetto a quelle dei precedenti metodi, ma non è facile verificarla visto che i linguaggi usati nei testi possono essere vari, pieni di sfumature, per questo è bene far si che tale procedimento sia fatto da degli esperti.

L'altro svantaggio è il costo computazionale, soprattutto se si analizzano centinaia di migliaia di testi il processo di stima può essere molto oneroso nonostante si utilizzano tecniche come quella di bagging.

Capitolo 3

Integrazioni al metodo di Hopkins e King

Si propongono ora alcune strategie differenti nell'applicare il metodo di Hopkins e King con l'obiettivo di migliorare le sue prestazioni.

Leggendo il paper di Amati et al. (2014) (documento in cui si utilizza l'Information Retrieval per velocizzare l'algoritmo, vedi anche Amati et al. (2008)) si è colto un suggerimento: non utilizzare tutti gli stem presenti nei testi che si hanno a disposizione, ma fare una scelta accurata degli stessi e usarne solo un sottoinsieme. In questo modo si concentra tutta l'analisi su pochi stem significativi dai quali si aspetta di trarre un'informazione più precisa. Ovviamente la riduzione deve essere giustificata e tale per cui in un secondo momento sia possibile valutare la bontà della scelta fatta.
Le modifiche proposte sono due: nel primo caso si usa il metodo degli alberi di classificazione CART di Breiman et al. (1984) e della Random Forest di Breiman (2001) da una parte per estrarre gli stem che hanno una maggiore potere predittivo e dall'altra per assegnare i pesi ideali a ciascuno di essi nel processo di bagging.
La seconda modifica invece si basa sulla sostituzione delle parole con un sinonimo per cercare di ridurre il numero di stem e aumentare il riconoscimento degli word stem profiles caratteristici di ciascuna categoria.

3.1 Random Forest

La modifica qui proposta è quella di scegliere e pesare gli stem tramite un metodo di classificazione individuale come il Random Forest. In pratica, invece di utilizzare questo metodo per fare classificazione, lo si utilizza per assegnare a ciascuno stem un punteggio, dopo di che si procede con il metodo di Hopkins e King sia utilizzando come insieme degli stem profiles quello generato dagli stem che hanno un potere predittivo maggiore, sia assegnando

a ciascuno stem un peso sulla base di alcuni indici che il metodo di classificazione usa per la costruzione dell'albero, come quello di Gini. Per cui si sfruttano questi indici per stilare la graduatoria degli stem, partendo da quello che ha un potere predittivo maggiore.
Si noti quindi che un metodo individuale non sia del tutto inutile per fare classificazione di tipo aggregato, ma viene utilizzato con un secondo fine per cercare di incrementare la bontà del classificatore aggregato.

I pesi vengono assegnati in modo proporzionale ad uno degli indici. Usando come indice il decremento medio dell'indice di Gini, sia G_j il valore di tale indice assegnato allo stem j, il peso sarà dato da:

$$p_j = \frac{G_j}{\sum_{j=1}^{J} G_j} \tag{3.1}$$

È logico che questo lavoro dovrà esser fatto su tutti gli stem che sono nel training set o comunque quelli che vengono utilizzati per fare la stima delle categorie nel metodo di Hopkins e King.

Si osservi che i pesi sono calcolati in modo strettamente dipendenti dalle categorie, cosa che non accade con l'assegnamento previsto da Hopkins e King. In quel caso i pesi erano assegnati o in modo uniforme oppure in base alle frequenze degli stem, quindi in modo indipendente dalle categorie.

Si ricorda che i classificatori ad albero si costruiscono tramite processi di separazione binomiale, per cui ad ogni bivio, in base ai valori assunti dalle variabili, si procede verso un ramo oppure un altro fino al raggiungimento dei nodi foglia. Per decidere come splittare nel modo ottimale un nodo nei suoi due rami si introducono degli indici (di purezza o impurezza). Lo split ottimale sarà definito come quello che, fra tutti gli split possibili, genera il massimo incremento o decremento dell'indice stesso. L'indice di Gini è l'indice di impurezza comunemente utilizzato che nel caso di classificazione in due gruppi è definito da $G = p(1-p)$, quindi pari alla varianza della variabile aleatoria con distribuzione Bernoulli di parametro p, che indica la probabilità di appartenenza ad una delle due categorie.
Nel caso multiclasse come indice di Gini si potrebbe usare il seguente: $G = 1 - \sum_{j=1}^{J} p(j)^2$, in cui con $p(j)$ si indica la probabilità di appartenenza alla classe j.

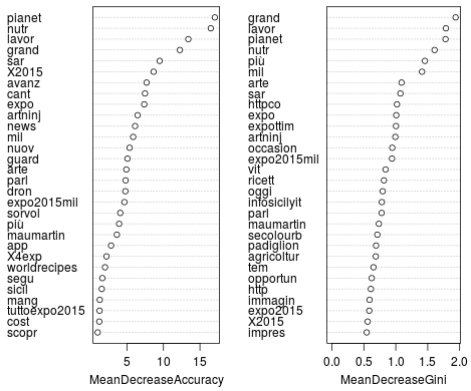


Figura 3.1: *Esempio di un gruppo di stem ordinati in base agli scores assegnati dall'algoritmo Random Forest.*

3.1.1 La scelta degli stem e l'assegnamento dei pesi

Stabilire a priori quale sia il numero di stem ideale da utilizzare non è banale, si procede allora a fare delle prove sapendo che con un numero eccessivamente basso si avrebbe una eccessiva perdita di informazioni, dal lato opposto invece non si vogliono utilizzare tutti gli stem. Solitamente si tende a fissare un threshold e ad escludere gli stem che compaiono poche volte e quelli che compaiono troppe volte (quindi escludere quelli che compaiono sotto il *threshold*% delle volte e sopra il (*1-threshold*)% delle volte). In questo caso si vuole fare comunque un'ulteriore selezione. Si procederà partendo da un gruppo poco numeroso che man mano, seguendo la classifica stabilita dal Random Forest in base all'indice (come quella mostrata nella figura 3.1), verrà arricchito di nuove parole nella speranza di arrivare ad un punto in cui si trova una numerosità ideale che consenta di ottenere un'accuratezza massima e migliore di quella ottenuta usando tutti gli stem. Quindi si procederà tramite cross-validazione per fare la scelta ideale.

Riassumendo si hanno due nuove proposte: un nuovo sistema di pesi e una selezione degli stem che hanno un maggiore potere predittivo. In questo caso si propone di stilare la classifica e assegnare i diversi pesi tramite il valore del decremento medio dell'indice di Gini.
In realtà sia i nuovi pesi, sia l'ordine degli stem più potenti possono esser fatti anche con altri strumenti, bisognerà trovare quello migliore, che meglio si adatta al problema che si sta affrontando.

3.2 Sinonimi

Si vuole ora presentare un ulteriore approccio che avrà sempre lo scopo di migliorare le prestazioni del metodo di Hopkins e King, prima di procedere con la descrizione si forniscono alcuni esempi introduttivi che chiariscono da dove è nata l'idea del cambiamento proposto e perché, almeno in via teorica, dovrebbe migliorare le prestazioni.

Si considerino le seguenti frasi, che potrebbero essere tweet di due utenti differenti: (a) *"He is courageous and proud"* e (b) *"He is brave and gallant"*. Si osserva che le due frasi dicono la stessa cosa, nonostante vengano utilizzate parole differenti, esprimendo un giudizio positivo nei confronti di una terza persona. Supponiamo ora che la frase (a) faccia parte del training set, mentre la frase (b) sia un nuovo tweet e che il profilo *brave + gallant* non faccia parte del training set. Di conseguenza avremo che $P(S = \textit{brave + gallant}|D = j) = 0$ per qualsiasi j, per cui la frase (b) non contribuirà ad aumentare la percentuale di testi positivi (si riveda la formula (2.6)), nonostante il testo sia chiaramente positivo.

Allora l'idea è quella di lavorare su tutto il corpus di testi, ancor prima di procedere con la costruzione del classificatore, sostituendo quando possibile i sinonimi con un loro rappresentante. Quindi, supponendo che *courageous* sia il rappresentante della classe {*audacious, brave, braw, courageous, dauntless, fearless, hardy, intrepid, unfearing*} e che *proud* lo sia per la classe {*chivalrous, dashing, gallant, knightly, lofty, majestic, proud*}, la frase (b) avrà uno stem profile uguale a quello della frase (a), in questo caso le frasi diventano proprio identiche. Di conseguenza quando si procederà alla valutazione di $P(S = \textit{brave + gallant})$ questa sarà più alta rispetto a prima e fornirà a sua volta un contributo maggiore alla categoria di tweet positivi, come è giusto che sia.
Con questo piccolo esempio si osserva che procedendo alla sostituzione con il sinonimo rappresentante di ogni classe il metodo dovrebbe migliorare, infatti molti stem profiles che in precedenza non fornivano contributo a nessuna classe, ora invece danno un significativo contributo. Si noti anche che questo metodo aiuta anche a far si che l'assunzione (2.7) sia verificata, infatti essa prevede che stem profiles come quelli della frase (b) non rimangano fuori dal training set e che quindi le parole ed il linguaggio utilizzato nel training set siano rappresentative del linguaggio usato da tutta la popolazione di utenti di cui si stanno studiando i testi. Inoltre l'analista avrà un maggior controllo sulle parole utilizzate e potrà verificare con più facilità che essa sia realmente verificata. Infatti nell'applicazione di questo metodo si procede anche alla costruzione di un vocabolario dei sinonimi accessibile a posteriori.
Per poter applicare questo metodo si deve disporre di un vocabolario tramite il quale si procede alla ricerca dei sinonimi. È quindi importante che questo vocabolario sia ben fatto e sia chiaro. Un problema non banale è riuscire

a trovare un vocabolario che si possa interfacciare con il software che si sta utilizzando.

Come si può immaginare questo procedimento non è così semplice, il problema sta proprio nella complessità del linguaggio, per cui sapere quando e se sia possibile procedere con la sostituzione è una cosa molto delicata che richiede attenzione. Ma proprio a causa della complessità si devono stabilire delle regole che ci consentano di raggiungere l'obiettivo senza alterare il contenuto dei testi. Si procede quindi alla descrizione del modus operandi e delle regole formulate.
È molto importante notare che questo procedimento di sicuro non peggiorerà i risultati ottenuti con il metodo classico. Infatti se in precedenza uno stem profile non contribuiva ad incrementare le percentuali di una certa categoria (oppure incrementava quella della categoria sbagliata), ora o viene riconosciuto e quindi porterà ad un miglioramento, oppure non verrà riconosciuto e continuerà a non portare alcun miglioramento. In pratica l'uso dei sinonimi porta ad essere più dettagliati e meno approssimativi, per cui o si continua a fare stime sbagliate oppure si otterranno dei valori più vicini a quelli veri.

3.2.1 Regole di sostituzione

Sostituzione univoca. Si è già parlato della complessità del linguaggio, uno dei problemi legato al discorso dei sinonimi è che una parola presenta diversi tipi di sinonimi che, in contesti differenti, assumono significati completamente diversi. Tanti aggettivi per esempio sono participio passato di alcuni verbi, di conseguenza non si può procedere alla sostituzione automatica di un verbo con un aggettivo o viceversa. Allora si stabilisce che la sostituzione sia fatta solo ed esclusivamente se la classe di sinonimi in cui compare un certo termine è unica. Un caso è proprio quello della parola *brave* che è sia aggettivo sia verbo, mentre la parola *gallant* è usato sia come aggettivo sia come nome. Quindi con le frasi (a) e (b) non si è in grado di fare una sostituzione univoca ed automatica, sarà necessario andare a leggere la frase, capire il contesto e poi sostituire, cosa che un algoritmo non è in grado di fare. È ovvio che esistono invece tantissime parole che appartengono ad un'unica classe di equivalenza, un esempio è l'aggettivo *fascinating*.

Seppure le sostituzioni non potranno che migliorare le performance del metodo, si precisa che questa scelta ne limita le potenzialità, tantissime parole infatti hanno diversi tipi di sinonimi. Una strada alternativa potrebbe essere quella di assegnare una priorità alle classi, ovvero fare sempre la sostituzione procedendo alla ricerca dei sinonimi tra i nomi, poi nella lista degli aggettivi, poi in quella dei verbi e via dicendo. Sarebbe bene approfondire questo discorso con degli esperti del linguaggio che possano dare delle linee guida su questo aspetto.

Parole insostituibili. Come si può immaginare, soprattutto se si stanno trattando testi scritti sui social network, spesso sono presenti parole scritte in modo scorretto, sia con errori grammaticali involontari, ma anche scritte in modo non corretto appositamente per enfatizzare il concetto che si vuole esprimere. Questo rappresenta un limite non solo in un qualsiasi metodo che applica la sentiment analysis, ma anche e soprattutto se si vuole utilizzare questo approccio. Infatti se invece di scrivere "*mi piace molto*" si scrivesse "*mi piace mooooltoooo*" sarebbe impossibile per un qualsiasi software riconoscere la parola *molto* e quindi nel nostro caso andare a cercare la lista dei suoi sinonimi. Allora in questi casi non si può fare altro che lasciare invariate queste parole. Questo vale in tutti i casi in cui non si è in grado di ricavare una lista di sinonimi perché la parola non è presente nel vocabolario.

Scelta del rappresentante della classe. Una volta che si ha tutta la classe di sinonimi di una certa parola (compresa la parola stessa), si deve procedere alla scelta del rappresentante. Questo passaggio potrebbe sembrare semplice, in generale basterebbe scegliere un elemento qualsiasi della classe, in realtà per quanto è stato detto nei punti precedenti la scelta è un punto molto importante e delicato.
In questo caso si propone il seguente metodo: si sceglie la parola più frequente in tutto il corpus di testi che si ha a disposizione e si procede a sostituire tutte le altre parole della classe con questo rappresentante. Ovviamente la scelta ideale sarebbe quella di usare come rappresentante la parola più frequente nel linguaggio, se non si è in grado di reperire una classifica delle parole più frequenti di una certa lingua si usa il criterio sopra descritto.
Anche stavolta si potrebbero fare altre scelte che siano più adatte al problema che si sta trattando, qui si propone la cosa più semplice o comunque quella che sembra più giusta.

Capitolo 4

Applicazione: EXPO Milano 2015

L'esposizione universale che si terrà a Milano dal 1 maggio al 31 ottobre 2015 è un grande evento che viene organizzato ogni quattro anni in una città del mondo, durante questo evento centinaia di nazioni si riuniscono in un unico posto per celebrare un certo tema, per esempio l'evento di quest'anno è intitolato *Nutrire il Pianeta, Energia per la Vita*. Per cui gli occhi di tutto il mondo sono puntati verso il paese che lo organizza, inoltre 20 milioni di persone sono attese a Milano in questi sei mesi.
Proprio perché è un evento importante ha un forte impatto sia sulla popolazione ospitante sia sui visitatori che arrivano dall'estero. Da mesi in tutti i telegiornali, giornali, social network e siti internet si continua a parlare di Expo, principalmente per quelle che sono le problematiche ad esso legate come per esempio il ritardo dei lavori oppure tutti gli affari illeciti che girano attorno all'organizzazione dell'evento e costruzione e allestimento del sito in cui si terrà l'esposizione.

A questo punto si vogliono sfruttare il metodo e le integrazioni finora descritte per capire come un evento di tale portata sia percepito dalle persone (sia italiani che stranieri) e successivamente indagare e capire quali sono gli aspetti che si possono migliorare oppure quale sia l'impatto di certi eventi reputazionali che potrebbero ledere l'immagine dell'Expo. Questo tipo di analisi è stata fatta su due fronti: sono stati scaricati i tweet in italiano per studiare il sentiment espresso dal paese ospitante dove l'evento ha sicuramente più eco, poi sono stati scaricati anche i tweet in lingua inglese per capire come l'evento viene invece percepito dall'esterno. Ovviamente la condizione ideale sarebbe stata quella di riuscire a scaricare i tweet in base alla geo-localizzazione, ma questo servizio non sempre è utilizzato dagli utenti ed inoltre ci sono stati dei problemi nel riuscire a scaricare i tweet in base a questo parametro. Quindi per discriminare le due popolazioni di utenti ci

si è affidati completamente alla lingua usata nel testo: si confronterà quindi l'opinione del popolo che parla dell'evento in italiano e del popolo che invece usa la lingua inglese.
In tutti e due i casi si parla di dati scaricati ancor prima dell'inizio dell'esposizione, di conseguenza si analizzeranno quelle che sono le percezioni che le persone hanno sulla base di quello che viene raccontato loro in TV, sui social network o altri mezzi di diffusione e alla luce di tutti gli scandali che si sono susseguiti.
Sono stati scaricati dati da Twitter relativi a 3/4 settimane a cavallo tra i mesi di febbraio e marzo 2015.
Si precisa fin da subito che il reperimento dei dati è molto difficoltoso, o meglio risulta molto complicato avere dei dati gratuitamente. Non avendo nessun tipo di budget sono stati utilizzati tutti gli strumenti gratuiti a disposizione.

Tutte le analisi che seguiranno sono state fatte usando il software di analisi statistica *R* (R Core Team (2014a) su Windows 7, mentre R Core Team (2014b) su Ubuntu 14.10).

Sentiment e Opinion analysis Fino ad ora si è sempre parlato solo ed esclusivamente di sentiment analysis, ma gli stessi metodi in realtà possono essere usati per una classificazione testuale più generica. La sentiment analysis può essere usata in due modi differenti. Da una parte si possono individuare e registrare tutti gli eventi che possono aver avuto un impatto positivo o negativo sulle persone e verificare questi sospetti tramite la sentiment analysis. Dall'altra parte invece si può fare un lavoro di esplorazione, molto interessante e strettamente legato alla sentiment analysis, che è chiamata *opinion analysis*. Questo tipo di analisi diventa molto più precisa della sentiment analysis, poiché si pone l'obiettivo di indagare sui motivi che hanno portato al sentiment. Quindi è sempre un'analisi del sentimento, ma molto più precisa e dettagliata. Solitamente viene affiancata alla sentiment proprio perché da un quadro più completo.
Mentre nella sentiment analysis le categorie sono fissate a priori in base a quello che si vuole sapere, nell'opinion analysis la maggior parte delle volte si ricavano nella fase di tagging manuale, quindi solo leggendo parte dei testi. Ovviamente dipende da quello che si sta studiando, in alcuni casi si conosce talmente a fondo il problema e l'opinione che le persone hanno che alcune categorie vengono fissate a priori e poi integrate con altre che arrivano appunto dalla lettura dei dati.
Si consideri il caso in cui si voglia conoscere l'opinione dei consumatori sul nuovo I-Watch. Fare sentiment analysis vuol dire verificare se il nuovo dispositivo piace o meno ai consumatori, fare opinion analysis invece vuol dire capire perché piace o perché non piace (per esempio la qualità e le prestazioni

del display, della batteria, del processore ecc.).
Si procede quindi facendo in primo luogo una sentiment analysis per capire se l'evento in generale è percepito in modo positivo o meno e successivamente si indagano i motivi di tale sentiment facendo un'opinion analysis.

4.1 Analisi e considerazioni preliminari

4.1.1 Scelta delle categorie e tagging manuale

Come si è detto si vogliono seguire i due approcci della sentiment e dell'opinion analysis. L'approccio legato alla sentiment analysis risulta abbastanza facile da capire, le categorie utilizzate infatti sono quelle classiche: *positivo*, *negativo*, *neutro*, *no-opinion* e *off-topic*. L'intento in questo caso è quello di sapere se le persone hanno un impressione positiva o negativa dell'Expo e di tutto quello che gira attorno a questo evento. È bene però descrivere in modo dettagliato quali sono i tipi di testi che sono stati catalogati come appartenenti a ciascuna di queste categorie. Si ricorda infatti che il tagging manuale risulta essere la pratica più difficile e lunga di tutto il metodo, ma è anche la più importante e delicata. Per avere un quadro completo quando si analizzeranno i risultati si specificano le assegnazioni che sono state fatte (vedi tabella 4.1).

Per quanto riguarda l'opinion analysis invece si vorrebbe andare più a fondo e capire i motivi che hanno spinto le persone a dare un giudizio positivo o negativo per poter capire se ci sono degli aspetti che possono essere migliorati. Si riporta una tabella esplicativa (tabella 4.3) con tutte le categorie sulle quali si vuole indagare. Come si può notare a tal proposito le categorie che non sono né positive né negative sono state accorpate per puntare l'attenzione sull'obiettivo prima descritto.

Quindi si osserva che se ci fossero alte percentuali nelle categorie 6 e 8, per esempio, si potrebbe intervenire per risolvere gli eventuali problemi o ridurre i danni. Mentre per la categoria 7 si ha poca capacità di intervento immediato.

È bene osservare fin da subito che fare il tagging manuale quando si trattano testi scritti sui social network (Twitter, Facebook, Google+, ecc.) presenta una serie di difficoltà che però devono essere affrontate fissando fin da subito delle regole, in modo da essere sempre coerenti. L'approccio risulta infatti differente in base ai social netwoks che si stanno analizzando. Per i testi scaricati da Twitter si decide come trattare i retweet, su Facebook invece si ha a che fare con i commenti che spesso sono legati al post (che non

Tabella 4.1: *Sentiment analysis. Categorie su cui si vorrebbe indagare.*

Sentiment Analysis	
POSITIVO/ NEGATIVO	Opinione riferita ad Expo, o ad altri eventi collegati all'esposizione, con messaggi positivi/negativi chiari ed espliciti.
NEUTRO	In questo caso l'utente non si sbilancia tra negativo e positivo, per esempio se si esprimono delle perplessità sulla riuscita di EXPO (senza piega negativa), oppure se si esprimono delle aspettative nei confronti di Expo: *speriamo vada bene!*, ecc.
NO-OPINION	Il testo parla di Expo, ma non esprime nessuna opinione (informazioni varie sull'evento, domande chiare, ecc.). Quindi qui ci saranno tutti i testi scritti da un utente istituzionale, in particolare dall'account ufficiale, ma anche da tutti quegli account ad esso collegati che danno informazioni generiche sull'esposizione.
OFF-TOPIC	Tutto ciò che va aldilà dell'evento o che comunque non ha niente a che vedere con l'esposizione. Quindi anche tutto quello che non si capisce, parole singole, discorsi non chiari perché fuori dal loro contesto, frasi incomplete e troncate. Tutti i casi di indecisione.

Tabella 4.3: *Opinion analysis. Categorie su cui si vorrebbe indagare.*

Opinion Analysis		
POSITIVO	1	Sentiment positivo nei confronti di Expo espresso in modo esplicito ed argomentato, ma del tutto **generico**: *ce la faranno...*, *non vedo l'ora di vedere...*, ecc.. Tutto quello che c'è di positivo non riconducibile ad altre classi.
	2	**Avanzamento e conclusione dei lavori** in tempo (padiglioni conclusi e consegnati ecc.).
	3	**Adeguatezza delle infrastrutture** (hotel, strade, trasporti pubblici, ecc.), non di competenza dell'organizzazione Expo.
	4	Giudizi positivi su **eventi che citano Expo** (legati o meno ad Expo o che espongono logo Expo).
	5	**Effetti positivi che Expo avrà sulla società**, che sia in Italia o all'estero, tra cui la Corporate Social Responsability.
NEGATIVO	6	Pareri negativi sull'evento, molto **generici**: pareri sull'estetica dei padiglioni, critica a designer o architetti, costo del biglietto eccessivo, critica agli sponsor e ad altre scelte fatte non condivise. Tutto quello che c'è di negativo non riconducibile ad altre classi.
	7	**Corporate Misconduct**: corruzione, tangenti, tutto ciò che è illecito oppure è percepito come tale.
	8	**Bad Management**: cattiva gestione, tutto ciò che è lecito ma gestito male, quindi ritardi nei lavori e cattiva gestione operativa (coperture già rovinate), accessibilità per disabili, contratti di lavoro, problemi nella vendita dei biglietti (anche critica per il ritardo nella vendita dei biglietti per più giorni).
	9	**Non adeguatezza delle infrastrutture** (hotel, strade, trasporti pubblici, ecc.), non di competenza dell'organizzazione Expo.
	10	Giudizi negativi su **eventi che citano Expo** (legati o meno ad Expo o che espongono logo Expo).
ALTRO	11	Tutto quello che è **no-opinion**, **neutro** o **off-topic**. Tutti i casi di forte indecisione. Semplice condivisione e approvazione del post pubblicato (in tal caso non si ha certezza a cosa o chi ci si sta riferendo): *Bello!*, *Fantastico*, *Mia piace*, ecc.; discorsi non chiari perché fuori dal contesto; info eventi legati o meno ad Expo (che non esprimono un sentimento); info sull'evento: orari, biglietti, info logistiche e servizi vari; commenti il cui messaggio non è chiaro o incompleto o testi mal codificati.

sempre è noto) oppure che sono parte di un discorso di cui non si conosce nulla.
Un'altra particolarità è il fatto che tra i testi ci sono tanti off-topics, ma non sempre si è in grado di riconoscerli. Qui risulta molto importante il supporto degli esperti, che sanno esattamente se l'argomento trattato nel testo risulta essere in qualche modo collegato ad Expo oppure no. Nei casi che si studieranno e approfondiranno di seguito c'è stata una collaborazione con il Dipartimento di Ingegneria Gestionale del Politecnico di Milano, in particolare con la professoressa Marika Arena, che ha fornito le indicazioni necessarie per costruire le categorie sopra elencate e per fare un hand coding che sia il più corretto possibile. È ovvio che sarebbe stato molto meglio se il tagging fosse stato fatto solo ed esclusivamente da più esperti, quindi si deve tenere ben presente che il training set è sicuramente migliorabile.
Questa grande numerosità di off-topics porta a dover etichettare a mano un numero molto maggiore di testi rispetto a quelli che venivano indicati da Hopkins e King per poter avere così una numerosità adeguata di testi in ciascuna categoria come richiesto dall'assunzione (2.7). Questo fattore è anche strettamente legato all'argomento che si sta affrontando, tutti usano gli hashtag usati per la ricerca (elencati di seguito) anche quando non si sta parlando dell'Expo: c'è chi è in cerca di popolarità, chi si fa pubblicità o fa pubblicità a terzi. Se si fosse affrontato un altro tipo di problema, anche scaricando i dati dei social network, questo fenomeno sarebbe sicuramente ridotto.

Per comprendere a fondo le scelte e in parte le analisi è importante conoscere come è fatto Twitter, a tale scopo in appendice A è presente una breve descrizione.

4.1.2 Download dei dati

Viste le difficoltà descritte è stato usato il pacchetto *twitteR* (Gentry, 2015) che consente di scaricare i testi ed altre informazioni relative ai tweet in base alla presenza di determinate parole chiave. È stata molto utile la funzione che consente di scaricare e salvare su un file sql (tramite l'uso del pacchetto *RSQLite*, Wickham (2014)) tutti i dati evitando di scaricare più volte gli stessi tweet, ma ci sono in realtà tanti problemi legati al recupero dei dati in questo modo. Questi problemi sono dovuti principalmente a delle limitazioni imposte da Twitter, secondo le quali non è possibile reperire dati del passato più in la di una settimana. Di conseguenza si è provveduto a scaricare i dati giorno dopo giorno per un mese intero (dal 17 febbraio al 16 marzo 2015).

Per essere chiari questi sono i comandi del pacchetto *TwitteR* che sono stati utilizzati:

- *search_twitter_and_store("#expo2015 OR #expo2015milano OR #expomilano2015 OR #expomilano","store_expo",lang="it")*

- *search_twitter_and_store("#expo2015milan OR #expomilan2015 OR #expomilan","store_expo_en",lang="en")*

I dati comprendono i seguenti campi:

- ID del tweet
- Testo del tweet
- Data e ora di invio
- Nome utente che lo ha mandato
- Nome dell'eventuale utente che riceve il tweet
- N° di retweet ricevuti
- N° di preferiti ricevuti
- Se il tweet è un retweet di un altro utente
- Eventuali informazioni di geo-localizzazione (latitudine e longitudine)

Negli studi che seguiranno è stata fatta solo un'analisi dei testi, ma si potrebbero usare tutte le precedenti informazioni per renderla più completa.

4.1.3 Valutazione dell'accuratezza

Prima di procedere con l'analisi dei due dataset introduciamo due indici che useremo per valutare la bontà dei metodi che verranno utilizzati. Per fare tale valutazione si procede tramite cross-validazione, dividendo l'insieme di testi etichettati manualmente in training e testing set. In particolare viene scelto il 75% casuale dei testi etichettati manualmente per il training set, il restante 25% per il testing set. Il training set sarà usato per costruire il classificatore che a sua volta verrà utilizzato per fornire le stime delle percentuali di testi del testing set in ciascuna categoria. Si indicano con T_j la percentuale vera di testi nel testing set nella categoria j, mentre con E_j la percentuale stimata dal classificatore. Allora si introduce il seguente indice che è comunemente usato in letteratura e in particolare dai due autori Hopkins e King:

$$I_d = \sqrt{\frac{\sum_{j=1}^{J}(T_j - E_j)^2}{J}} \tag{4.1}$$

Questo indice descrive in modo classico la distanza tra i due vettori di probabilità, è sempre positivo e la stima sarà migliore tanto più il suo valore si avvicina allo 0.

L'indice che segue, usato nella tesi di laurea di Branca (2014), invece mette a confronto due categorie, per questo è spesso usato nei casi in cui si studia il classico sentiment *positivo-negativo*. Si denota con $T_{j|j\vee\tilde{j}}$ la vera percentuale di testi nella categoria j condizionata al fatto che il testo stia in una delle categorie tra j e $\tilde{j}$, mentre con $E_{j|j\vee\tilde{j}}$ la stima della probabilità che un testo stia nella categoria j sempre condizionata al fatto che il testo stia in una delle categorie tra j e $\tilde{j}$ e si costruisce il seguente indice:

$$I_c = \log\frac{T_{j|j\vee\tilde{j}}}{E_{j|j\vee\tilde{j}}} - \log\frac{T_{\tilde{j}|j\vee\tilde{j}}}{E_{\tilde{j}|j\vee\tilde{j}}} \tag{4.2}$$

Si osserva che può assumere valori positivi e negativi ed è nullo se i valori stimati sono pari a quelli veri, quindi in generale la stima sarà tanto migliore quanto il valore assoluto dell'indice si avvicina allo 0 . Dal punto di vista interpretativo il segno di I_c sarà positivo se si ha una sovrastima della categoria $\tilde{j}$ e quindi una sottostima della categoria j, sarà negativo nel caso opposto. Quindi sarà particolarmente utile per evidenziare e confrontare eventuali flussi da una categoria ad un'altra. Proprio per le sua caratteristiche di basarsi solo su due categorie, questo indice da meno informazioni sulle prestazioni complessive del classificatore. Come al solito dipende da quale sia lo scopo del lavoro: se si è interessati a costruire un classificatore che valuti bene la massa di probabilità in solo due categorie allora l'indice è perfetto. Si ricorda infatti che spesso dal punto di vista pratico alcune delle categorie vengono inserite per migliorare le prestazioni del classificatore, ma dal punto di vista interpretativo non serviranno a niente. Questo ci ha portati inizialmente a fare la distinzione tra tweet neutri e quelli che non esprimevano opinione, in tutti e due i casi si parla di Expo ma non si da ne un giudizio positivo ne uno negativo.

4.1.4 Il pacchetto *ReadMe*

Per applicare il metodo si userà il pacchetto *ReadMe* (Hopkins et al., 2013). Questo pacchetto si interfaccia con Python e prendendo in ingresso i testi, le etichette e i vari parametri come il numero di sottoinsiemi, la loro numerosità e i pesi degli stem, restituisce la distribuzione di probabilità in ciascuna delle categorie. Quindi, sia applicando il metodo di Hopkins e King descritto nel

capitolo 2 sia le variazioni che sono state proposte nel capitolo 3, alcuni parametri rimarranno fissi e non cambieranno mai in tutte le strategie e i dataset che si analizzeranno, altri invece saranno scelti e valutati caso per caso. Si elencano di seguito i parametri fissi.

Threshold all'1%. Come è stato detto, non tutti gli stem, che compaiono nel training set verranno utilizzati per costruire il classificatore, ma si impongono dei limiti. Infatti se lo stem compare raramente oppure troppo spesso non sarà utile a distinguere i vari word stem profiles, quindi è preferibile escludere quello stem. In tutti i casi che studieremo è stato imposto un threshold dell'1%, pari a quello di default del pacchetto.

Numero di sottoinsiemi pari a 300. Quando è stato descritto il metodo di Hopkins e King, è stata introdotta la tecnica di bagging, ovvero l'insieme complessivo degli stem (che soddisfano il threshold imposto precedentemente) viene suddiviso in tanti gruppi, a quel punto si applica il metodo sui singoli gruppi producendo delle stime di $P(D)$ per ognuno di essi e solo alla fine si procede a fare una media aritmetica di tali valori. Anche in questo caso è stato usato il valore di default di 300 sottoinsiemi, considerando il fatto che più sono e meglio è, tale valore sembra ragionevole ed accettabile.

Numero di parole in ogni sottoinsieme. Il suggerimento di Hopkins e King per stabilire quale sia il numero di stem che vanno a formare i singoli sottogruppi è quello di procedere per cross-validazione. In seguito procederemo ad eseguire le analisi considerando due valori: nel primo caso verranno selezionate 15 parole (valore di default), nel secondo caso si ridurrà tale numero a 8 stem per gruppo. Non si ritiene opportuno utilizzare un numero maggiore di stem poiché si stanno analizzando dei tweet, quindi testi che hanno al massimo 140 caratteri, il che vuol dire circa 10 o 12 stem significativi (ovvero al netto di stopwords, link, emotions, spazi vuoti e punteggiatura) per ogni frase.

4.1.5 Definizione delle strategie

Nelle sezioni che seguiranno si procederà ad eseguire la sentiment e l'opinion analysis sui dataset in lingua italiana ed inglese. L'obiettivo è quello di confrontare i risultati che si ottengono con il metodo già esistente di Hopkins e King con quelli che si ottengono utilizzando le integrazioni che sono state presentate, verificando se queste portano a dei miglioramenti, garantendo quindi un classificatore più preciso.
Per semplificare l'esposizione si introducono tre strategie che riflettono le proposte di cambiamento fatte nei capitoli precedenti, quindi il metodo è sempre quello di Hopkins e King, cambierà invece il modo di assegnare i pesi o il numero di stem da usare nei sottoinsiemi.

L'analisi verrà fatta prima usando 15 stem per ogni sottogruppo, poi invece 8 stem per sottogruppo.

- **Strategia 1**: è il modello base di default del pacchetto ReadMe, vengono utilizzati tutti gli stem del training set, esclusi quelli che non passano il limite imposto dal threshold. Ad ogni stem è assegnato un peso binomiale, quindi si ha che $p_j = \frac{\frac{1}{q_j(1-q_j)}}{\sum_{j=1}^{J} \frac{1}{q_j(1-q_j)}}$. La stima di q_j è fatta ovviamente in modo empirico sulla base dei testi che si hanno a disposizione.

- **Strategia 2**: in questo caso si usa il decremento medio dell'indice di Gini, assegnato a ciascuno stem nella procedura del Random Forest, sia per assegnare i pesi a ciascuno di essi, sia per creare la classifica. In particolare si analizzeranno i casi in cui si prendono i primi 20 stem, poi si integrano con altri 20 per un totale di 40 stem, poi 60, 80, arrivando infine ad usarli tutti.

- **Strategia 3**: in questo caso si usa il decremento medio dell'indice di Gini solo per creare la classifica (si analizzeranno anche qui i casi cui si prendono i primi 20 stem, poi 40, 60, 80, arrivando infine ad usarli tutti). I pesi invece saranno uguali per tutti gli stem, vengono quindi assegnati in modo uniforme.

Si noti che nella terza strategia il caso in cui si utilizzano i pesi uniformi e tutti gli stem è il caso base di Hopkins e King. Quindi sarà importante confrontare la strategia 1 e l'ultimo caso della 3 con tutto il resto, che rappresenta la parte innovativa del lavoro di tesi.
Nella tabella 4.5 si propone uno schema riassuntivo.

Random Forest: impostazione dei valori. Per determinare il decremento medio dell'indice di Gini per tutti gli stem del training set, come è stato detto, è stato utilizzata la procedura Random Forest. Si considerano i soli testi taggati manualmente, quindi quelli del training set, e tramite il pacchetto *tm* (Feinerer, 2014) si costruisce la matrice $\mathbf{B}$, per cui $B_{i,k}$ vale 1 se lo stem k è presente nel testo i, 0 altrimenti. Dopo di che si costruisce un nuovo dataframe in cui una colonna rappresenta la categoria associata al testo e tutte le altre sono dedicate ai singoli stem, ogni riga è invece dedicata ad un testo. Ora tramite il pacchetto *randomForest* (Breiman and Cutler, 2014) si procede con la classificazione costruendo 500 alberi. Dall'output della classificazione si estraggono gli indici di Gini e li si utilizzano opportunamente come sopra specificato.

Tabella 4.5: *Schema delle strategie.*

	Pesi	N° stem per sottoinsieme
Strategia 1	binomiali	TUTTI
Strategia 2	proporzionali al decremento medio dell'indice di Gini	- 20 - 40 - 60 - 80 - TUTTI
Strategia 3	uniformi	- 20 - 40 - 60 - 80 - TUTTI

Si segnala che, a causa di problemi di codifica di alcuni caratteri presenti nei testi, non è stato possibile assegnare a tutti gli stem del training il relativo peso. Infatti questi problemi hanno impedito un match tra gli stem usati dalla funzione *ReadMe* e questi ultimi. Si è deciso di assegnare a tali stem un peso nullo. Questo porterà ovviamente all'indebolimento della procedura e alla perdita di parte dell'informazione che questi stem portavano.

4.2 Twitter: testi in italiano

La prima ricerca è stata fatta selezionando il popolo di Twitter che usa la lingua italiana. Le parole chiave utilizzate per il download dei dati sono state: *#expo2015, #expo2015milano, #expomilano2015* ed *#expomilano*. Come si può notare non è stato utilizzato l'hashtag *#expo* perché troppo generico e spesso usato quando si fa riferimento ad una qualsiasi esposizione, di conseguenza ci sarebbero stati dei dati completamente fuori tema. Come è stato già detto avere dei tweet in cui il tema non è quello dell'esposizione universale di Milano è molto frequente, per questo è stato deciso di limitarne il numero eccessivo già in partenza non usando quest'hashtag.
Si è proceduto al download dei tweet giornalmente dal 17 febbraio all'11 marzo 2015, ottenendo così un totale di 76 380 tweet.
Per le nostre analisi è stato fatto un controllo a priori sugli utenti escludendo coloro che hanno twittato più di 10 volte al giorno circa, quindi è stata imposta una soglia di 300 tweet complessivi nel periodo di studio in questione. Questa scelta deriva dal fatto che ci sono molti profili interamente dedicati ad Expo che inoltrano, tramite i retweet, una grande quantità di informazioni che quindi risultano duplicate e prive di sentiment. Giusto per fare un esempio, in una fase intermedia, su 40 000 tweet 7000 sono stati inviati da

un unico utente i cui tweet sono stati quindi rimossi dal dataset. In questo modo si è arrivati a ridurre il dataset a 49 403 tweet.

Nella tabella 4.6 sono riportate le caratteristiche principali.

Tabella 4.6: *Analisi descrittiva del dataset in italiano.*

Numero di tweet	49 403
• numero dei quali sono retweet di altri tweet	36 277
Numero di utenti diversi che hanno twittato	19 812
Numero medio di volte che un tweet è stato retwittato	18 circa
Numero totale url (link esterni)	52 223
Numero medio di url per tweet	1 circa
Numero di volte che compaiono i vari hashtags usati per la ricerca	• #expo2015: 46 829 • #expomilano: 1422 • #expomilano2015: 1272 • #expo2015milano: 310

Si può notare che l'hashtag più usato è il primo, mentre gli altri aggiungono ben poca informazione. Significativo è invece l'alto numero di retweet e link esterni.

4.2.1 Risultati del tagging manuale

Come prima cosa si è proceduto con il tagging manuale. Sono stati selezionati 700 testi da tutto il corpus di 49 403 testi in modo del tutto casuale, sono stati letti ed etichettati. Questa procedura è stata fatta contemporaneamente nei due sensi: sentiment e opinion analysis. Quindi nel caso in cui un tweet veniva considerato positivo o negativo ci si chiedeva il perché e si assegnavano le categorie dell'opinion analysis, se invece il testo era neutro, non esprimeva un'opinione oppure era off-topic veniva catalogato come *altro*.

Questa è una scelta che è stata fatta per rendere la procedura veloce, ma i testi potrebbero essere scelti anche in modo mirato, proprio per rendere le categorie complete (pratica comune nel caso di categorie poco numerose).
Si ripete ulteriormente che la cosa fondamentale non è la numerosità totale dei testi etichettati o le numerosità dei singoli gruppi (non deve esserci per forza un campione statisticamente rappresentativo e neanche un numero di

testi identico in ciascuna categoria), ma solo ed esclusivamente che tutte le categorie siano linguisticamente ben rappresentate dai testi scelti.

In figura 4.1 si riportano i risultati del tagging manuale per quanto riguarda le categorie della sentiment analysis.
Per quanto riguarda l'opinion analysis invece si sono ottenuti i valori presentati in figura 4.2.

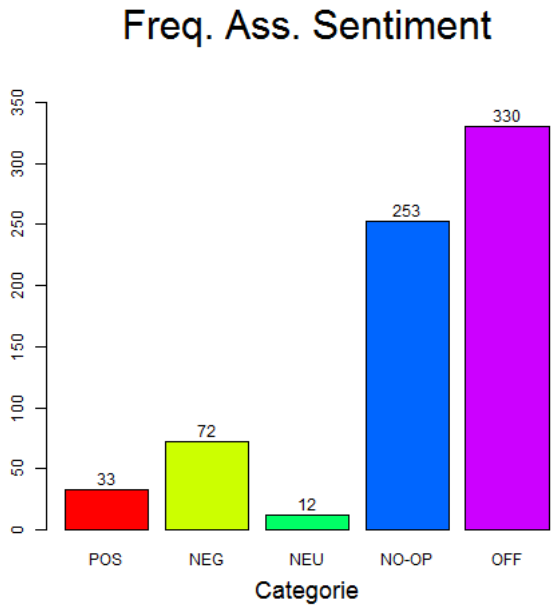


Figura 4.1: *Numerosità delle categorie di sentiment analysis dopo il tagging manuale di 700 testi casuali in italiano.*

Poiché alcune delle categorie elencate precedentemente risultano poco numerose e si ha bisogno che ogni categoria abbia un discreto numero di testi, si procede con l'accorpamento di alcune di esse. Come è stato spiegato nei capitoli precedenti, l'accorpamento delle categorie è una procedura necessaria in questi casi, il grande svantaggio è la perdita di informazioni dettagliate, per cui si dovranno indagare diversamente le cause dei vari sentiment. Per evitare di fare ciò si potrebbe continuare ad etichettare altri testi fino al raggiungimento di una numerosità accettabile, ma questo richiede molto più tempo. Quindi ci si pone nel caso intermedio optando per l'accorpamento di alcune di esse.

Per quanto riguarda la sentiment analysis si è deciso di accorpare i tweet neutri con quelli che non esprimono opinione. La decisione su quali categorie dell'opinion analysis accorpare è stata presa sotto consiglio della professores-

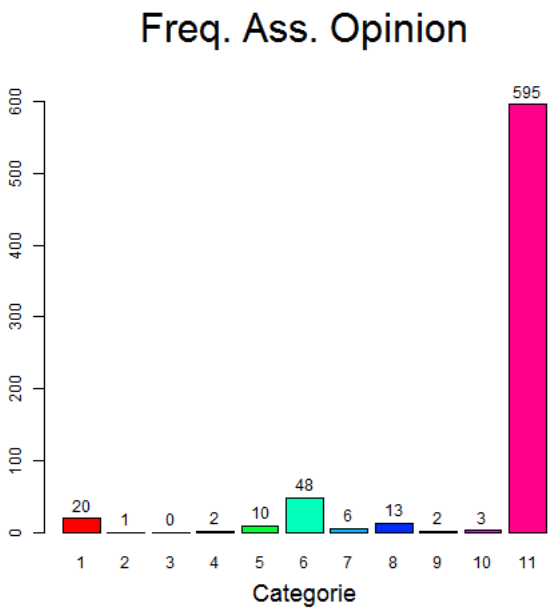


Figura 4.2: *Numerosità delle categorie di opinion analysis dopo il tagging manuale di 700 testi casuali in italiano.*

sa Arena, arrivando alla sintesi presentata nella tabella 4.7. In particolare per quanto riguarda il sentiment positivo sono state accorpate tutte le categorie lasciando sola quella riguardante gli effetti positivi che l'evento avrà sulla società, arrivando ad avere due categorie finali. Per quanto riguarda il sentiment negativo sono state unite la categoria riguardante gli affari illeciti con quella relativa alla cattiva gestione, tutto il resto che esprime giudizi negativi generici sull'Expo o sugli eventi ad esso legati sono stati invece messi assieme in un'altra categoria. Anche nel caso del sentiment negativo si hanno quindi due categorie che lo descrivono. Di conseguenza complessivamente per l'opinion analysis si hanno cinque categorie finali.

Si riportano nella figura 4.3 le numerosità della sentiment analysis, mentre in figura 4.4 quelle dell'opinion analysis.

Tabella 4.7: *Categorie finali dell'opinion analysis dei testi in italiano dopo aver accorpato alcune di esse per scarsa numerosità.*

Opinion Analysis: post accorpamento		
	Cat.	Descrizione
POSITIVO	1	Giudizi positivi generici su Expo; avanzamento e conclusione dei lavori; adeguatezza delle infrastrutture; giudizi positivi su eventi legati ad Expo.
	2	Effetti positivi che Expo avrà sulla società.
NEGATIVO	3	Giudizi negativi generici; Non adeguatezza delle infrastrutture; giudizi negativi su eventi legati ad Expo.
	4	Corporate misconduct; bad management.
ALTRO	5	Tutto quello che è no-opinion, neutro o off-topic (rimane invariata).

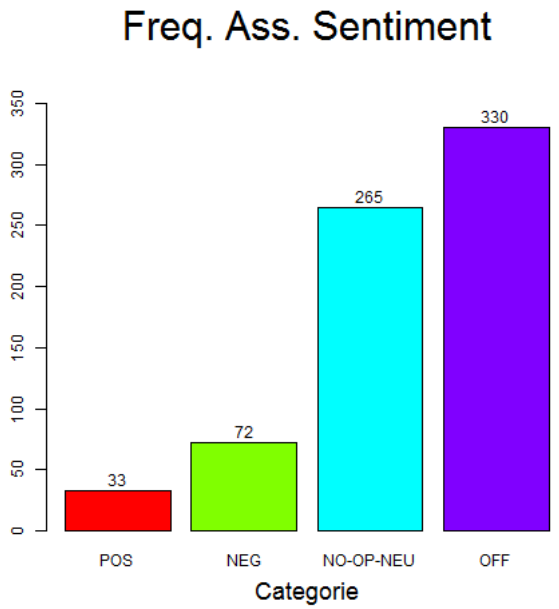


Figura 4.3: *Numerosità delle categorie di sentiment analysis dopo il tagging manuale di 700 testi casuali in italiano e dopo aver accorpato i tweet neutri a quelli senza opinione.*

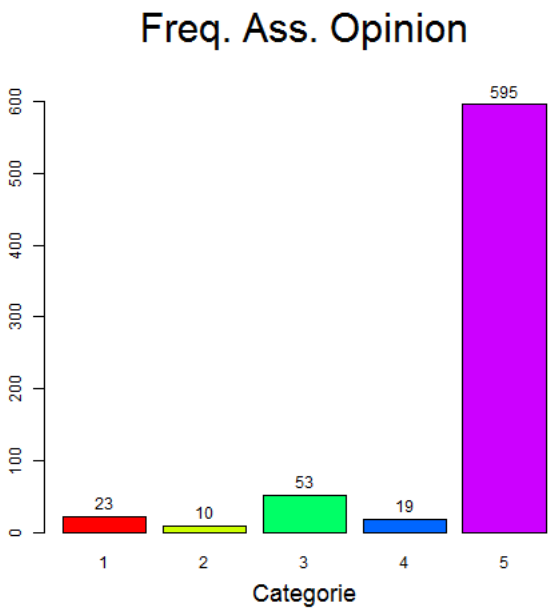


Figura 4.4: *Numerosità delle categorie di opinion analysis dopo il tagging manuale di 700 testi casuali in italiano e dopo aver accorpato le categorie poco numerose.*

4.2.2 Preprocessing dei testi ed individuazione dei modelli

Si procede a questo punto con la pulizia di tutto il corpus di testi che si hanno a disposizione. Si eliminano quindi gli spazi vuoti in eccesso, si riduce il testo ad avere solo lettere minuscole e si eliminano sia le stopwords italiane (vedi Appendice C) sia i collegamenti esterni ai siti. Si eliminano inoltre altre parole che fin da subito sappiamo essere poco informative dal punto di vista del sentiment, queste sono: *2015, expo, expo2015, expomilano, expomilano2015, expo2015milano*. Quindi si eliminano tutte le parole usate per la ricerca e altre come *rt* che sono tipiche nei testi di twitter perché indicano che il testo è stato ritwittato e non è scritto direttamente dall'utente che lo ha inviato.
Sempre parlando del caso in cui un utente (a) ritwitta il tweet di un altro utente (b), oltre alla parola RT compare in modo automatico anche il nome dell'utente b che aveva originariamente scritto il tweet, quindi, facendo un esempio, i dati a disposizione ci informano che l'utente a ha scritto *"RT @utente-b: #Expo2015: la disperata corsa contro il tempo per salvare la faccia WWWURLWWW via @utente-c, @utente-d"*. *@utente-b* diventa quindi parte integrante del testo che si andrà ad analizzare e si è concluso che questa parte non debba essere eliminata, infatti potrebbe diventare uno stem significativo per individuare l'appartenenza del testo ad una delle categorie.

Questa scelta potrebbe essere discutibile, ma è sembrata comunque sensata. Un aspetto strettamente legato ai social network è la perdita di parte dell'informazione dovuta alla moda di scrivere le parole in modo abbreviato. Questa abitudine era già presente quando si scrivevano i classici sms, ma è ancora più presente ora, sopratutto su Twitter vista la limitazione imposta sul testo che deve avere al massimo di 140 caratteri. Molto usata è inoltre l'enfatizzazione del messaggio duplicando o triplicando le lettere sia finali sia centrali delle parole. Per questi motivi è molto importante cercare, fin che è possibile, di aggiornare gli strumenti che si useranno in modo da affrontare queste problematiche, ne è un esempio l'introduzione tra le stopwords di parole come *xché*, *xké*, ecc. In tanti altri casi però non si può avere un controllo. Questo in generale porterà ad una perdita di informazioni e quindi una sottostima di certi stem (se si scrive *"Mi piace moooltooo!!!"* lo stem *molt* non verrà riconosciuto e *moooltooo* sarà visto come uno stem a se stante). Quindi bisogna tener presente questo aspetto, per esempio Hopkins e King testano il loro metodo sui blog di politica, quindi ci si aspetta un linguaggio più grammaticalmente corretto e riconoscibile dalle funzioni e software che verranno usati.

Si è prestata la massima attenzione nell'assicurarsi che tutte le procedure abbiano l'effetto desiderato. L'ordine con il quale vengono applicate è infatti importante. Se si togliesse un apostrofo tra un articolo e una parola, questi ultimi due si condenserebbero in un'unica parola, bisogna invece togliere prima l'articolo (contenuto nelle stopwords) e poi l'apostrofo (che fa parte della punteggiatura).

Per ultimo si è proceduto ad effettuare lo stemming usando lo stemmer di Porter fornito dal pacchetto *tm* ed impostandolo ovviamente sulla lingua italiana. A questo punto vengono creati tutti i file necessari per poter utilizzare il pacchetto *ReadMe*.

4.2.3 Sentiment analysis

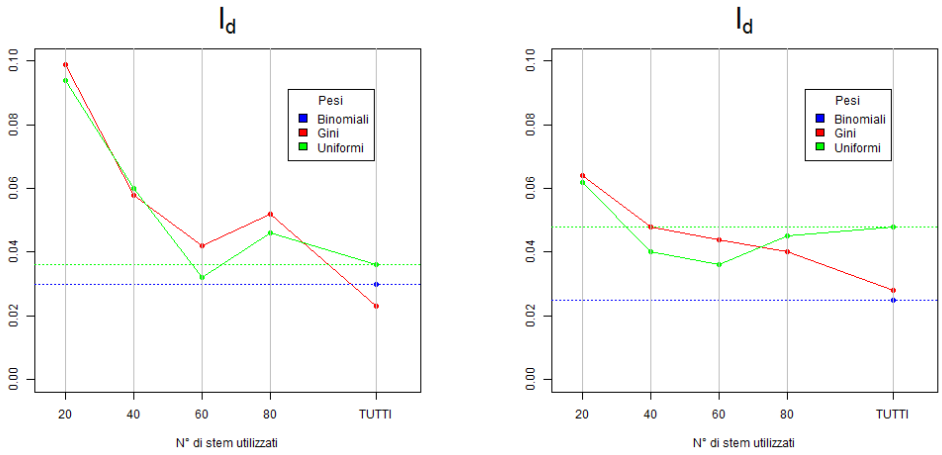
Si analizza ora la sentiment analysis, quindi il caso in cui le categorie sono *positivo, negativo, no-opinion/neutro e off-topic*. Si descrivono di seguito i risultati ottenuti con le diverse strategie, con le specifiche tecniche elencate precedentemente. Si ricorda che per la cross-validazione sono stati usati i 700 testi etichettati manualmente, in particolare il 75% casuale per il training set, mentre il restante 25% per il testing set.
Si riporteranno tutti e due gli indici, che verranno confrontati e si cercherà di capire quale sia il caso migliore e quindi se le modifiche apportate tramite Random Forest hanno portato ad un classificatore più efficiente. Il numero

totale di stem nel training set che rispettano i vincoli imposti dal threshold dell'1% è 111.

Tabella 4.8: *Sentiment analysis dei testi in italiano. Valori degli indici I_d e I_c al variare del numero di stem complessivamente usati e al numero di stem per sottoinsieme.*

		15 stem per sottoinsieme		**8 stem per sottoinsieme**	
	N°stem	I_d	I_c	I_d	I_c
Strategia 1	T	0.030	−0.026	0.025	0.025
Strategia 2	20	0.099	−1.755	0.064	−1.078
	40	0.058	−0.709	0.048	−0.544
	60	0.042	−0.924	0.044	−0.466
	80	0.052	−0.627	0.040	−0.196
	T	0.023	0.545	0.028	0.400
Strategia 3	20	0.094	−2.128	0.062	−1.024
	40	0.060	−0.496	0.040	−0.182
	60	0.032	−0.196	0.036	−0.271
	80	0.046	−0.250	0.045	−0.140
	T	0.036	0.015	0.048	−0.016

Si osservi come cambia l'indice I_d (si ricorda che ha la seguente espressione $I_d = \sqrt{\frac{\sum_{j=1}^{J}(T_j - E_j)^2}{J}}$) che valuta la distanza tra la massa di probabilità vera e quella stimata nel testing set. Sia dalla tabella 4.8, ma in modo più chiaro dalla figura 4.5a si osserva che le prestazioni migliori si hanno usando i pesi calcolati in base all'indice di Gini con tutti gli stem, ma l'errore non è tanto differente dal caso in cui si considerano i pesi binomiali.
Ora si procede a ridurre il numero di stem che andranno a formare i sotto-gruppi nella fase di bagging, si fissa tale numero a 8 stem. Questo potrebbe essere efficace visto che i testi che si stanno trattando sono dei tweet, che per loro natura sono costituiti da poche parole. Si riportano quindi sempre nella tabella 4.8 i valori degli indici che sono stati ricavati e analogamente a quanto fatto prima si riportano tali valori nei grafici in figura 4.5b. Alcuni dei valori cambiano, molti dei quali tendono ad abbassarsi, ma le strategie migliori rimangono comunque quelle di prima.

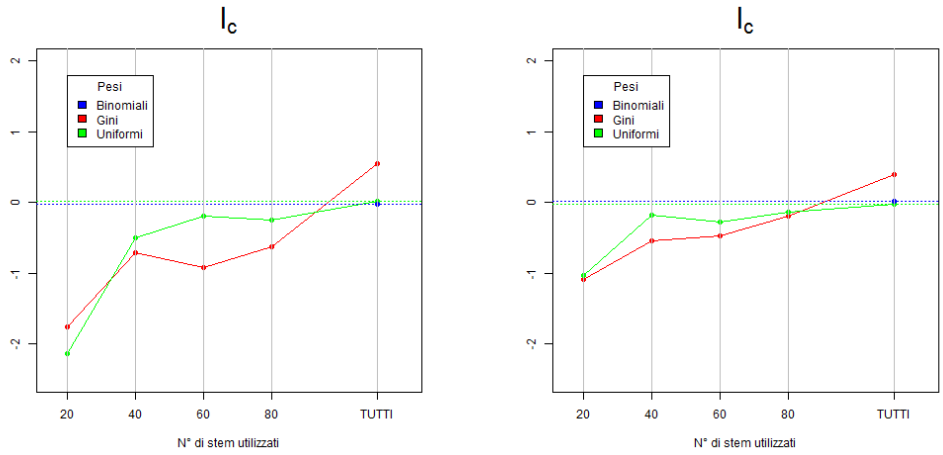


(a) *15 stem per sottoinsieme.* ***(b)*** *8 stem per sottoinsieme.*

Figura 4.5: *Confronto dell'indice I_d per la sentiment analysis del dataset in italiano. Le linee tratteggiate sono di riferimento poiché sono i risultati che si hanno con il metodo originale.*

Nella figura 4.6 sono riportati i valori di I_c ottenuti usando tutte le strategie confrontando la categoria dei testi positivi con quella dei testi negativi. Si ricorda che $I_c = \log \frac{Tc_j}{Ec_j} - \log \frac{Tc_{\tilde{j}}}{Ec_{\tilde{j}}}$, quindi $j = positivo$ e $\tilde{j} = negativo$). Usando 15 stem in ciascun sottogruppo, dalla figura 4.6a si nota che se si fosse interessati ad avere delle buone stime solo delle categorie positivo/negativo le strategie migliori sarebbero la prima e la terza, usando in entrambi i casi tutti gli stem. La stessa cosa succede in figura 4.6b, in cui si notano solo un leggero abbassamento dei valori. Non si notano infatti grosse differenze, in questo caso l'uso di 8 oppure 15 stem non ha prodotto grandi cambiamenti. Inoltre nella maggior parte dei casi l'indice I_c è negativo, quindi i rispettivi classificatori tenderanno a sovrastimare la categoria dei positivi e sottostimare quella dei negativi. Per molti aspetti potrebbe essere un difetto, infatti questo tipo di analisi sono solitamente fatte per capire se ci sono aspetti negativi e cercare di migliorarli, quindi una loro sottostima non è auspicabile.

In questa prima analisi non c'è stata una prova evidente che pesare diversamente gli stem o selezionarne un sottoinsieme porti a dei grossi miglioramenti, ma si è riusciti perlomeno ad eguagliare l'accuratezza del metodo con la strategia preesistente.



(a) *15 stem per sottoinsieme.* ***(b)*** *8 stem per sottoinsieme.*

Figura 4.6: *Confronto dell'indice I_c per la sentiment analysis del dataset in italiano. Le linee tratteggiate sono di riferimento poiché sono i risultati che si hanno con il metodo originale.*

Strategie ottimali. Per concludere si può dire che per avere un buon classificatore le strategie ottimali sono la prima e la seconda, usando tutti gli stem del training set e costruendo i sottoinsiemi per il bagging con 8 stem ciascuno. Infatti non solo i valori di I_d sono bassi e simili tra loro, ma se si fosse interessati a conoscere una stima precisa delle categorie *positivo* o *negativo* le due strategie garantirebbero un errore ridotto (il valore di I_c è basso). Certamente la terza avrebbe garantito un errore ancora più basso da questo punto di vista, ma il valore di I_d, a cui in generale si vuol dare maggior peso perché racchiude un'informazione completa di tutte le categorie, è quasi doppio.

Tra le strategie citate si decide di usare la prima. In figura 4.7 si mostra la relazione tra il valore vero delle frequenze e quello stimato, in particolare si costruiscono degli intervalli di confidenza al 95% per le frequenze medie con la tecnica bootstrap usando 100 ricampionamenti. Si precisa che il pacchetto *ReadMe* fornisce in modo automatico le stime degli standard error calcolati appunto con la tecnica bootstrap, di conseguenza gli intervalli di confidenza sono stati costruiti secondo la seguente formula:

$$IC_{0.95}(p(D_j)) = [\hat{p_j} - z_{0.95} * se \ , \ \hat{p_j} + z_{0.95} * se] \tag{4.3}$$

In particolare $\hat{p_j}$ è la stima puntuale, se è l'errore standard e $z_{0.95}$ è il quantile della normale standard, assumendo che con una numerosità alta il campione abbia distribuzione normale.

Si può vedere che i valori stimati sono vicinissimi a quelli veri e che in tutti e quattro i casi il valore ottimale (quindi quello vero) ricade all'interno dell'intervallo di confidenza.

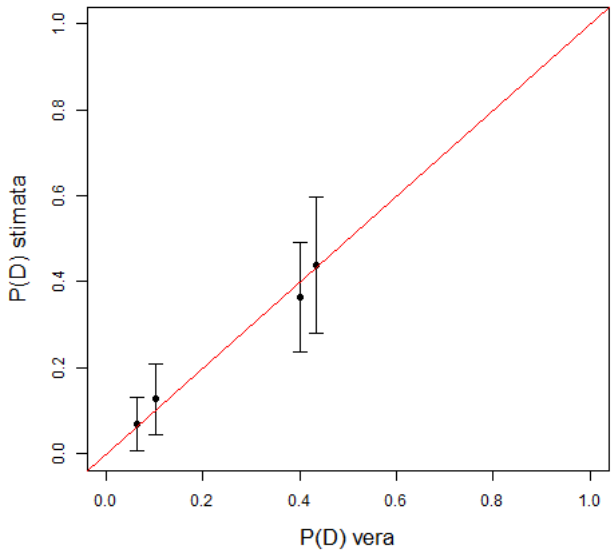


Figura 4.7: *Confronto tra le percentuali vere e quelle stimate di P(D) e* $IC_{0.95}$ *nel testing set. Sentiment analysis del dataset in italiano.*

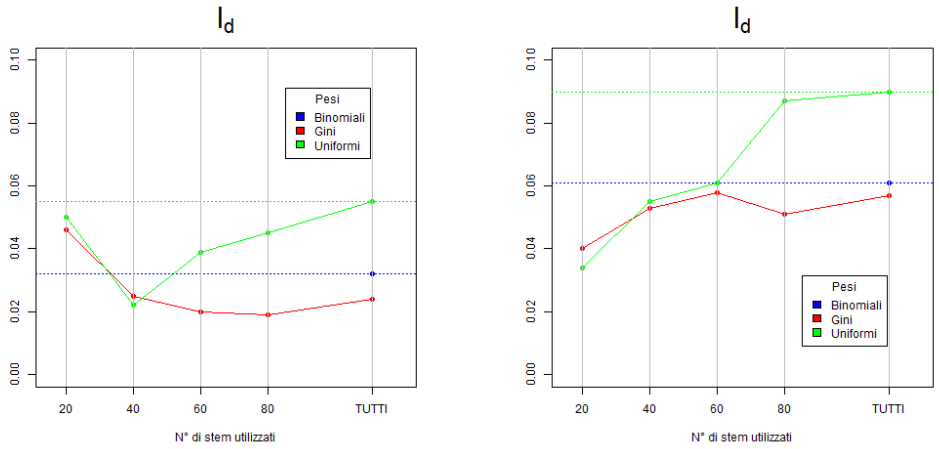
4.2.4 Opinion analysis

Si analizzano ora i risultati dell'opinion analysis. Si ricorda che alcune delle categorie originarie sono state accorpate, quindi si considera una classificazione con cinque categorie differenti, (vedi tabella 4.7). Analogamente a quanto fatto per la sentiment analysis si studiano le diverse strategie nel caso in cui si utilizzino 15 o 8 stem per sottogruppo.
Come si può osservare dalla tabella 4.9 si riportano i valori dell'indice I_d ottenuti, ma per ora non si valuta l'indice I_c poiché non ci sono due classi a cui si è particolarmente interessati e per cui si vorrebbe sbagliare di meno rispetto alle altre. Una volta analizzati tutti i risultati e scelto una strategia migliore quest'ultimo verrà calcolato per cercare di capire se ci sono particolari flussi di massa di probabilità da una categoria ad un'altra.

Dal grafico nella figura 4.8a si nota che, non solo usando i pesi proporzionali al decremento medio dell'indice di Gini ha portato ad un abbassamento

Tabella 4.9: *Opinion analysis dei testi in italiano. Valore dell'indice I_d al variare del numero di stem complessivamente usati e al numero di stem per sottoinsieme.*

		15 stem per sottoinsieme	**8 stem per sottoinsieme**
	N°stem	I_d	I_d
Strategia 1	T	0.032	0.061
Strategia 2	20	0.046	0.040
	40	0.025	0.053
	60	0.020	0.058
	80	0.019	0.051
	T	0.024	0.057
Strategia 3	20	0.050	0.034
	40	0.022	0.055
	60	0.039	0.061
	80	0.045	0.087
	T	0.055	0.090



(a) *15 stem per sottoinsieme.* ***(b)*** *8 stem per sottoinsieme.*

Figura 4.8: *Confronto dell'indice I_d per l'opinion analysis del dataset in italiano. Le linee tratteggiate sono di riferimento poiché sono i risultati che si hanno con il metodo originale.*

dell'errore nel caso in cui si utilizzano tutti gli stem, ma l'errore si riduce ulteriormente su si usa un numero inferiore di stem (il grafico mostra chiaramente la presenza di un minimo nella curva che riguarda la strategia 2). In particolare l'ottimo si ha usando 80 stem, si potrebbe indagare più a fondo e vedere se si possono ottenere valori migliori usandone poco meno o poco più di 80.

Osservando la figura 4.8b si nota che l'indice usando la strategia 2 è sempre inferiore rispetto ad usare la prima, ma il valore minimo lo si ha con la terza strategia, usando solo 20 stem. Ma se rapportiamo questi risultati ai precedenti si nota che in linea di massima i valori sono notevolmente cresciuti, il valore minimo viene quasi raddoppiato.

Strategie ottimali. Si può concludere che nel caso dell'opinion analysis è meglio usare 15 stem per ogni sottogruppo, selezionati tra quei 60/80 che hanno un decremento medio dell'indice di Gini maggiore e assegnando loro un peso proporzionale a tale indice.
In figura 4.9 si mostra il confronto tra il valore vero delle frequenze e quello stimato, costruendo anche stavolta degli intervalli di confidenza al 95% con la tecnica bootstrap usando 100 ricampionamenti e sempre usando la formula $IC_{0.95}(p(D_j)) = [\hat{p_j} - z_{0.95} * se\ ,\ \hat{p_j} + z_{0.95} * se]$. In tutti i casi il valore vero ricade all'interno dell'intervallo di confidenza.

Si osservi inoltre la figura 4.10 in cui si riporta l'andamento del segno di I_c calcolato mettendo a confronto tutte le cinque possibili categorie. Si ricorda che $I_c = \log \frac{T_{j|j \vee \tilde{j}}}{E_{j|j \vee \tilde{j}}} - \log \frac{T_{\tilde{j}|j \vee \tilde{j}}}{E_{\tilde{j}|j \vee \tilde{j}}}$.
Come si può vedere la categoria 1 viene sottostimata rispetto a tutte le altre categorie, infatti il valore di I_c è negativo ogniqualvolta $j = 1$, questo vuol dire che la stima ottenuta per la categoria 1 sarà inferiore rispetto a quella vera e la massa mancante è stata distribuita alle altre categorie. Per quanto riguarda le altre categorie si osserva in particolare che la categoria 2 è quella che viene sovrastimata rispetto a tutte le altre.
Quindi, in base a quale siano le categorie alle quali si tiene di più e per cui si vorrebbero delle stime migliori rispetto ad altre, si potrebbero dare dei giudizi positivi o meno sul classificatore.

Confronto trasversale. Facendo il punto della situazione, fino ad ora si potrebbe dire che il nuovo sistema di pesi e selezione di un gruppo di parole tramite l'indice di Gini ha portato dei buoni risultati. Si noti inoltre che si hanno notevoli differenze nel momento in cui si combinano gli effetti del

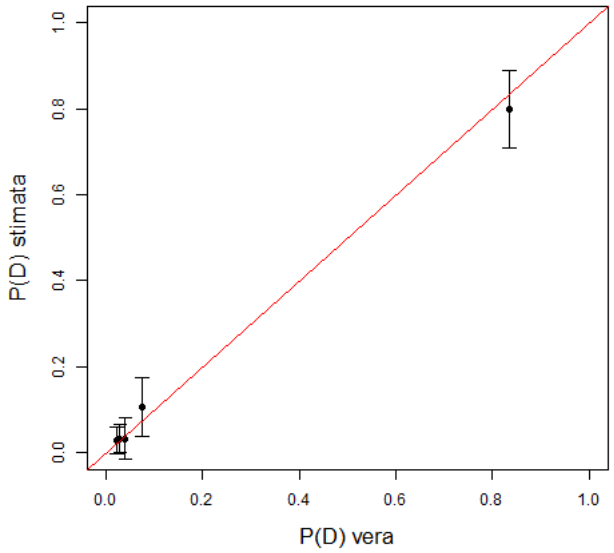


Figura 4.9: *Confronto tra le percentuali vere e quelle stimate di $P(D)$ e $IC_{0.95}$ nel testing set. Opinion analysis del dataset in italiano.*

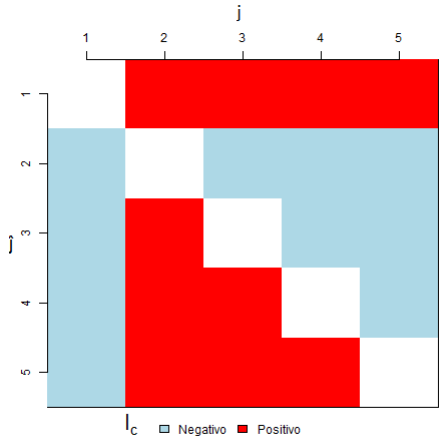


Figura 4.10: *Matrice dei valori dell'indice I_c, sull'asse delle ordinate l'indice j, mentre sulle ascisse l'indice $\tilde{j}$. Opinion analysis del dataset in italiano.*

Random Forest con la scelta ottimale del numero di stem da assegnare ad ogni sottoinsieme. Mentre nella sentiment analysis passando da 15 a 8 stem i valori tendono a diminuire, nell'opinion analysis si ha l'effetto opposto con un notevole aumento di tali valori.

4.2.5 Analisi dei risultati

Tra le strategie migliori che sono state trovate per la sentiment analysis si sceglie la 1, ovvero quella esistente già prima di questa tesi. Si ricorda che tale caso è caratterizzato dal costruire il classificatore usando tutti gli stem che rispettano il vincolo del threshold e assegnando loro un peso di tipo binomiale. In particolare la performance era migliore nel caso in cui, nella fase di bagging, si assegnano 8 stem ad ogni sottoinsieme.
Usando il modello con questa strategia si vogliono stimare le percentuali di testi di tutto il corpus di testi che si hanno a disposizione che vengono assegnate a ciascuna categoria.

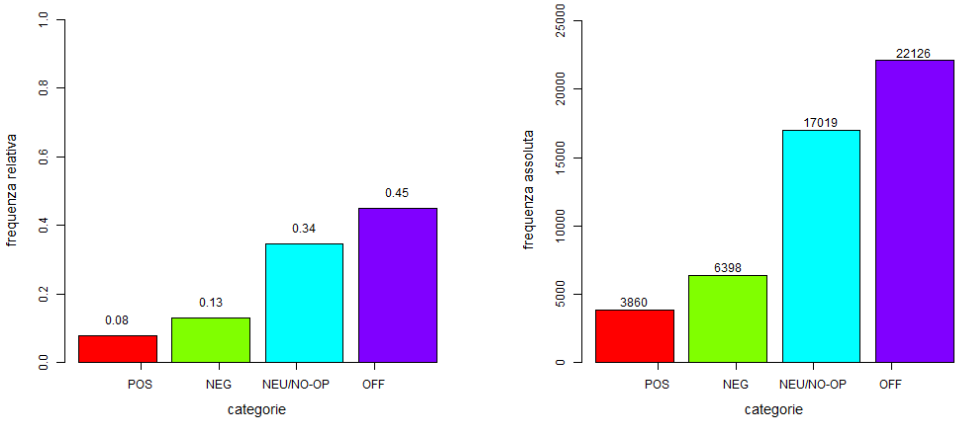


Figura 4.11: *Stime delle percentuali e dei valori assoluti di tweet in ciascuna categoria di sentiment del dataset in italiano.*

Dalla figura 4.11 si nota che la maggior parte dei testi è off-topic. Questo è un problema dello scaricamento dei dati dai social network, ognuno è libero di usare le parole che vuole e molti usano gli hashtag che fanno tendenza per attirare a se l'attenzione. La seconda categoria più numerosa risulta essere invece quella che racchiude i testi che non esprimono un'opinione. Si noti che in fase di tagging era stata notata l'alta numerosità di queste categorie visto che i testi erano stati presi in modo casuale, ma è bene non essere condizionati da questo fatto. Come è stato detto infatti la scelta dei testi da etichettare può essere fatta in vari modi, in base alle esigenze e i risultati

Tabella 4.10: *Categorie finali dell'opinion analysis dei testi in italiano dopo aver accorpato alcune di esse per scarsa numerosità.*

Opinion Analysis: post accorpamento		
	Cat.	Descrizione
POSITIVO	1	Giudizi positivi generici su Expo; avanzamento e conclusione dei lavori; adeguatezza delle infrastrutture; giudizi positivi su eventi legati ad Expo.
	2	Effetti positivi che Expo avrà sulla società.
NEGATIVO	3	Giudizi negativi generici; Non adeguatezza delle infrastrutture; giudizi negativi su eventi legati ad Expo.
	4	Corporate misconduct; bad management.
ALTRO	5	Tutto quello che è no-opinion, neutro o off-topic (rimane invariata).

non per forza rispecchiano le vere percentuali. Molti tweet erano descrittivi dell'evento, racchiudevano informazioni sui biglietti, sui metodi di acquisto e i rivenditori, molti altri tweet riguardavano eventi legati ad Expo oppure creati apposta in vista dell'evento, ma erano solo ed esclusivamente informazioni di servizio. Questo fattore potrebbe esser dovuto al fatto che ancora l'esposizione non è iniziata, quindi molti preferiscono non esprimersi o non hanno nessun interesse a riguardo.
C'è da considerare inoltre il fatto che durante il periodo che si sta analizzando non ci sono stati particolari scandali o comunque il tema Expo è stato poco trattato dai mass media. Questo porta al fatto che l'attenzione degli utenti Twitter non era sull'evento e si sono registrati quindi pochi tweet che esprimono un vero e proprio sentiment.
Osservando comunque i risultati si ha che l'8% del popolo Twitter appoggia l'evento o comunque ha un parere positivo, il 13% invece esprime un sentiment negativo.

Per investigare i motivi che hanno portato ad esprimere questi pareri si è proceduto con l'opinion analysis.
Si riporta di nuovo la tabella definitiva delle categorie dell'opinion analysis, ottenuta accorpando le diverse categorie poiché poco numerose (tab. 4.10).

La strategia migliore per ricavare delle stime più precise è risultata quella in cui si usavano i primi 80 stem della classifica stilata in base all'indice di Gini, assegnando ad ognuno di essi un peso proporzionale a tale indice (usando inoltre nella fase di bagging 15 stem per ogni sottogruppo). I risultati sono riportati nella figura 4.12.

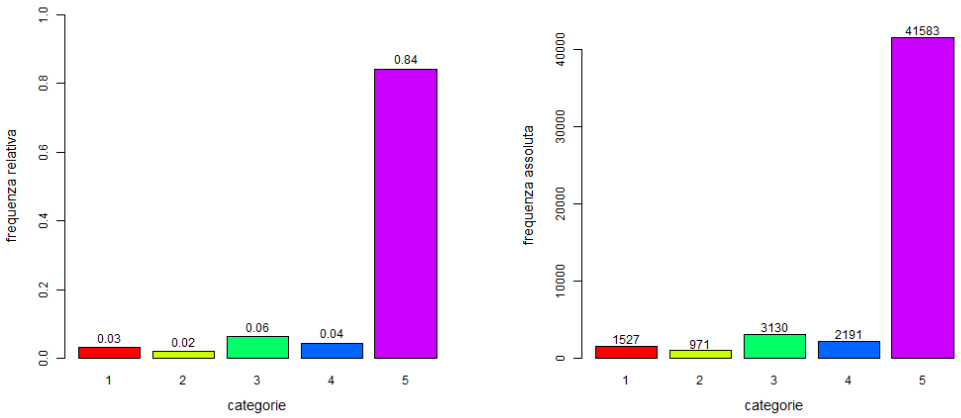


Figura 4.12: *Stime delle percentuali e dei valori assoluti di tweet in ciascuna categoria dell'opinion analysis del dataset in italiano.*

Si può osservare che per il 2% della popolazione di Twitter che usa la lingua italiana l'Expo è un evento positivo poiché avrà una serie di ripercussioni positive sulla società, in particolare su quella italiana. Mentre il 3% esprime un sentiment positivo generico.
Il sentiment negativo invece è per la maggior parte abbastanza generico o relativo alle scelte fatte da chi organizza l'evento. Questo potrebbe essere dovuto al fatto che, come detto prima, nel periodo in analisi non c'è stato nessuno scandalo particolare, quindi si potrebbe dire che è rimasto un sentiment negativo generico sull'evento dovuto ad avvenimenti precedentemente accaduti, legati principalmente agli scandali che si sono susseguiti, ma si ritiene che la categoria 4 (ora ferma al 4%) sia cresciuta anche dopo il periodo analizzato (nella metà di marzo, periodo fuori dall'intervallo di studio, c'è stato un ulteriore scandalo per tangenti e questo avrà portato all'ampliarsi del sentiment negativo per corruzione).

Particolare, ma non del tutto sorprendente, è il fatto che se, nell'opinion analysis, si vanno a sommare i contributi positivi tra loro, oppure quelli negativi, si ottengono dei valori leggermente differenti dalle stime della sentiment analysis. Lo stesso se nel caso precedente si sommassero le categorie *no-opinion/neutro* a quello relativo alla categoria 5. Questa differenza è legata al fatto che, nonostante i problemi della sentiment e opinion analysis siano strettamente legati tra loro e nel calcolo delle stime siano stati usati, ovviamente, gli stessi testi, i due sono affrontati come problemi diversi proprio perché le categorie sono tra loro differenti. La frammentazione delle categorie positivo e negative e l'unione delle altre due ha portato ad una matrice $\mathbf{P^h}(\mathbf{S}|\mathbf{D})$ differente, proprio perché i modelli usati sono del tutto differenti.

D'altronde si ricorda anche che in questo metodo non è possibile avere per ogni testo la relativa categoria assegnata, quindi dopo la sentiment analysis non si può trarre nessuna informazione per eseguire un'opinion analysis mirata, in particolare non si possono selezionare i testi catalogati come positivi per poi andare ad investigare più a fondo.

Analisi temporale. Per concludere l'analisi dei testi in italiano si riportano le stime delle proporzioni nelle categorie della sentiment e opinion analysis in 5 diversi istanti di tempo, prendendo tutti i tweet dei 5 giorni precedenti.

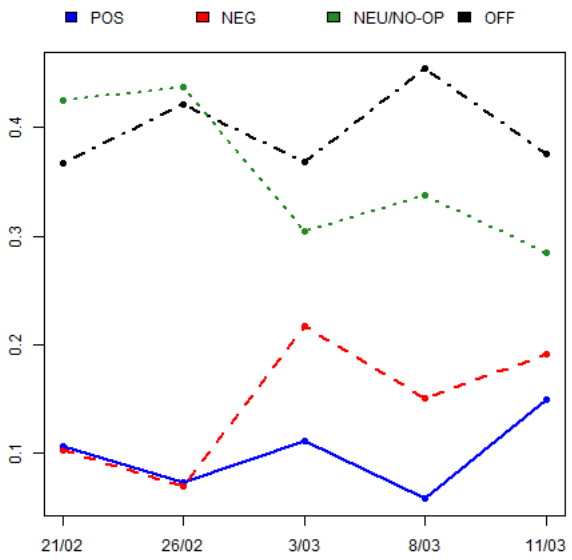


Figura 4.13: *Analisi temporale del sentiment del dataset in italiano.*

Dalla figura 4.13 si osserva che inizialmente le percentuali di tweet positivi e negativi sono state le stesse, successivamente la curva dei testi negativi si impenna, arrivando ad essere più del doppio rispetto a quelli positivi. Si noti come l'aumento dei tweet negativi ha portato ad una notevole riduzione delle percentuali di tweet che non esprimono nessuna opinione e neutri, mentre le percentuali di quelli off-topic oscillano un po' ma tendono a rimanere sempre e comunque molto alte.
Dalla figura 4.14, in modo particolare nel grafico sulla destra in cui è stato fatto lo zoom, si nota che il sentiment negativo di cui si parlava poco fa è di tipo generico, che riguarda anche le scelte fatte dagli organizzatori. Per cercare di capire a fondo cosa sia successo in quel periodo sono stati letti

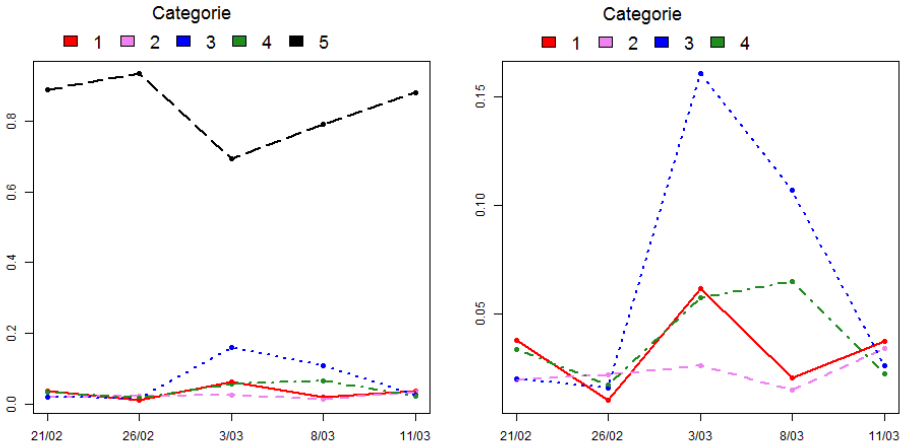


Figura 4.14: *Analisi temporale dell'opinion del dataset in italiano. A destra zoom sulle prime quattro categorie.*

alcuni testi riguardanti quell'intervallo di tempo e si è trovato che nei primi giorni di marzo Flavio Briatore ha lanciato una campagna di protesta perché il maiailino sardo non è stato inserito tra i cibi consentiti durante l'esposizione. Questa protesta ha avuta un certo eco, soprattutto nella popolazione sarda, ha scosso anche alcuni politici che si sono esposti, anch'essi sui propri profili Twitter, causando in questo modo questa crescita del sentiment negativo. Per cui, se gli organizzatori lo ritenessero importante, userebbero questa informazione per riparare al danno e placare le proteste prima che diventi un problema più serio.
Si osserva che nel giro di 15 giorni il fenomeno si esaurisce, infatti arrivati all'11 marzo le percentuali dei tweet nella categoria 3 ritornano più o meno pari a quelle del 26 febbraio.
Inoltre al contempo c'è anche una leggera crescita del sentiment positivo, ma non si è riusciti a capire quale sia stato l'evento o gli eventi ad esso legati.

Per riprendere il discorso che si faceva poco fa sull'incongruenza tra le due analisi, sentiment e opinion, si osservi come nell'ultimo giorno ci registrano dei trend opposti: nella sentiment analysis c'è un abbassamento delle percentuali associate alle categorie senza sentiment e un innalzamento di quelle positive e negative; nell'opinion analysis invece accade proprio il contrario, c'è una flessione positiva per quelle senza sentiment e una negativa per tutte le altre. Come si è detto questo aspetto è possibile e normale che esista, anche se si dovrebbe cercare di migliorare i due classificatori in modo che concordino su questi aspetti.

4.3 Twitter: testi in inglese

Si consideri ora il dataset in lingua inglese. Come è stato detto si vorrebbe capire se la popolazione che parla in inglese esprime un sentiment differente rispetto a quella italiana. In questo modo in realtà non possiamo essere certi che coloro che scrivono i tweet in inglese siano per forza stranieri, potrebbero essere anche italiani, ma l'unico modo disponibile in questo momento per poter selezionare una popolazione differente da quella italiana è quello della selezione tramite la scelta della lingua usata nel testo.
Per il download dei dati sono state utilizzate le stesse parole chiave del caso precedente (*#expo2015, #expo2015milano, #expomilano2015* ed *#expomilano*), alle quali sono state aggiunte *#expo2015milan, #expomilan2015* ed *#expomilan*, in cui è stato considerato il possibile uso di *milan* per indicare la città di Milano. Anche in questo caso il download dei dati è stato eseguito giornalmente dal 17 febbraio al 16 marzo 2015, scaricando così 23 068 tweet. Si osservi che nonostante l'intervallo preso in considerazione sia più ampio rispetto al dataset in italiano, il numero di testi è molto minore. Probabilmente al di fuori dell'Italia ancora non se ne parla tanto.
Facendo il controllo a priori sugli utenti ed escludendo coloro che nel mese che si sta analizzando hanno twittato più di 300 volte, si escludono 4993 tweet ottenendo così un dataset finale con 18 075 testi.
Nella tabella 4.11 sono riportate le caratteristiche principali e si ha che ancora una volta il primo hashtag è il più usato.

4.3.1 Risultati del tagging manuale

Anche stavolta si è proceduto con il tagging manuale. Sono stati selezionati 700 testi da tutto il corpus di 18 075 testi in modo del tutto casuale, sono stati letti ed etichettati. Questa procedura è stata fatta contemporaneamente nei due sensi, sentiment e opinion analysis. Si noti che 12 dei 700 testi non sono stati poi inseriti nel training set poiché contenevano solo link o solo caratteri strani che venivano codificati male, quindi sarebbero risultati vuoti dopo la fase di preprocessing. Quindi l'hand coded set è fatto da 688 testi.

In figura 4.15 si riportano i risultati del tagging manuale per quanto riguarda le categorie della sentiment analysis.
Per quanto riguarda l'opinion analysis invece si sono ottenuti i valori presentati in figura 4.16.

Poiché alcune delle categorie elencate precedentemente risultano poco numerose e si ha bisogno che ogni categoria abbia un discreto numero di testi, si procede dove possibile con l'accorpamento di alcune di esse. Per quanto riguarda la sentiment analysis anche stavolta si accorpano i tweet neutri con

Tabella 4.11: *Analisi descrittiva dataset inglese.*

Numero di tweet	18 075
• numero dei quali sono retweet di altri tweet	13 001
Numero di utenti diversi che hanno twittato	8653
Numero medio di volte che un tweet è stato retwittato	26 circa
Numero totale url (link esterni)	23 859
Numero medio di url per tweet	1.3 circa
Numero di volte che compaiono i vari hashtags usati per la ricerca	• #expo2015: 16 821 • #expomilano: 788 • #expomilano2015: 705 • #expo2015milano: 157 • #expomilan: 848 • #expomilan2015: 41 • #expo2015milan: 165

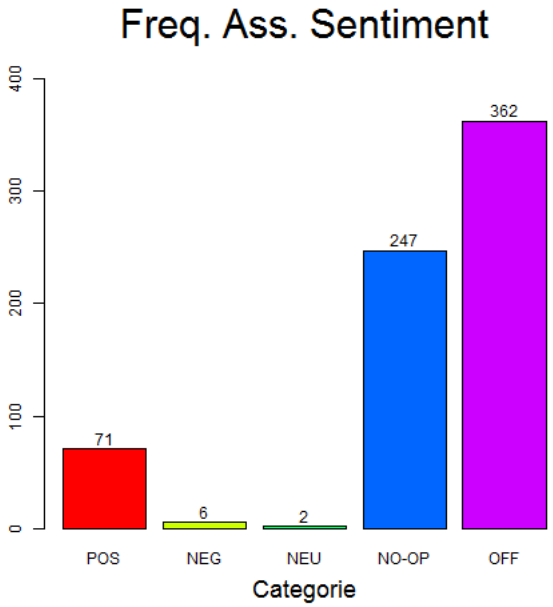


Figura 4.15: *Numerosità delle categorie di sentiment analysis dopo il tagging manuale di 688 testi casuali in inglese.*

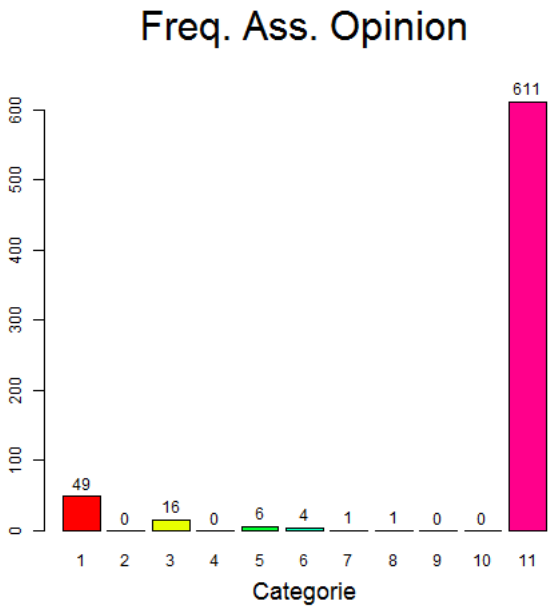


Figura 4.16: *Numerosità delle categorie di opinion analysis dopo il tagging manuale di 688 testi casuali in inglese.*

quelli che non esprimono un'opinione.
Per quanto riguarda l'opinion analysis le categorie sono composte da un numero eccessivamente basso di testi, per cui anche se si procedesse ad accorparne alcune non si avrebbero grandi miglioramenti. L'accorpamento infatti non va fatto in modo casuale col solo scopo di avere classi numerose, la suddivisione nelle categorie rimanenti deve continuare ad avere un senso nella fase interpretativa. Poiché in questo caso non è possibile trovare un rimedio si rinuncia all'opinion analysis.
Si riportano in figura 4.17 le numerosità della sentiment analysis dopo l'accorpamento. Si può notare che la categoria dei tweet negativi risulta poco numerosa, ma si decide di procedere comunque perché se la si accorpasse ad altre l'analisi perderebbe completamente il senso. La soluzione come al solito sarebbe quella di etichettare altri testi scelti in modo casuale o comunque cercare altri testi che esprimono un sentiment negativo fino al raggiungimento di una numerosità accettabile. Per mancanza di tempo si considerano tali valori accettabili, nella fase conclusiva dell'analisi si dovrà tener conto di una scarsa rappresentatività di questa categoria che potrebbe portare ad una stima inaccurata della massa di probabilità ad essa assegnata.

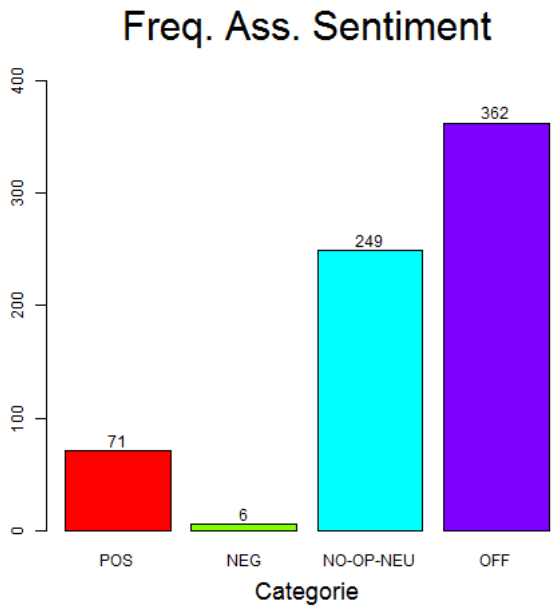


Figura 4.17: *Numerosità delle categorie di sentiment analysis dopo il tagging manuale di 688 testi casuali in inglese e dopo aver accorpato i tweet neutri a quelli che non esprimono un'opinione.*

4.3.2 Preprocessing del testo

Si procede quindi con il preprocessing del testo. Come fatto per i testi in italiano, si escludono le parole usate per la ricerca, le stopwords inglesi (vedi Appendice D) e la parola *rt*, mentre si lasciano nel testo i nicknames degli utenti che hanno scritto originariamente i tweet ritwittati. Dopodiché si procede con lo stemming del testo, sempre tramite il pacchetto *tm*, impostando la lingua inglese.

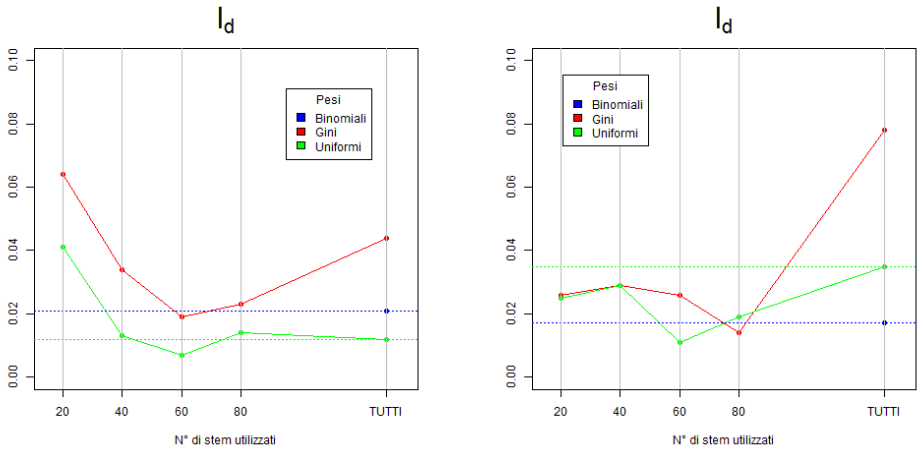
4.3.3 Sentiment analysis

Si analizza ora la sentiment analysis ricordando che, dopo l'accorpamento della categoria *neutro*, si è ridotta a 4 categorie. Si descrivono di seguito i risultati ottenuti con le diverse strategie, in particolare si riportano nella tabella 4.12 tutti e due gli indici. Ancora una volta si usano i testi taggati manualmente, quindi 688, divisi in training e testing set (rispettivamente il 75% e 25% casuali). Il numero totale di stem nel training set che varranno utilizzati per la stima è pari a 129.

Tabella 4.12: *Sentiment analysis dei testi in inglese. Valori degli indici I_d e I_c al variare del numero di stem complessivamente usati e al numero di stem per sottoinsieme.*

		15 stem per sottoinsieme		**8 stem per sottoinsieme**	
	N°stem	I_d	I_c	I_d	I_c
Strategia 1	T	0.021	0.866	0.017	1.126
Strategia 2	20	0.064	2.312	0.026	1.475
	40	0.034	1.741	0.029	1.654
	60	0.019	1.197	0.026	1.551
	80	0.023	1.338	0.014	0.960
	T	0.044	0.742	0.078	0.797
Strategia 3	20	0.041	1.901	0.025	1.631
	40	0.013	0.840	0.029	1.609
	60	0.007	0.720	0.011	0.998
	80	0.014	0.855	0.019	1.153
	T	0.012	1.030	0.035	1.339

Dalla figura 4.18a si osserva che usando 15 stem per sottogruppo, a parte nel caso in cui si utilizzano 20 stem per la stima, il modello con la strategia 3 ha una performance migliore rispetto a qualsiasi altra, in una qualsiasi configurazione. In particolare si ottiene un errore dello 0.7% se si usano i primi 60 stem nella classifica degli stem ordinata in base al decremento



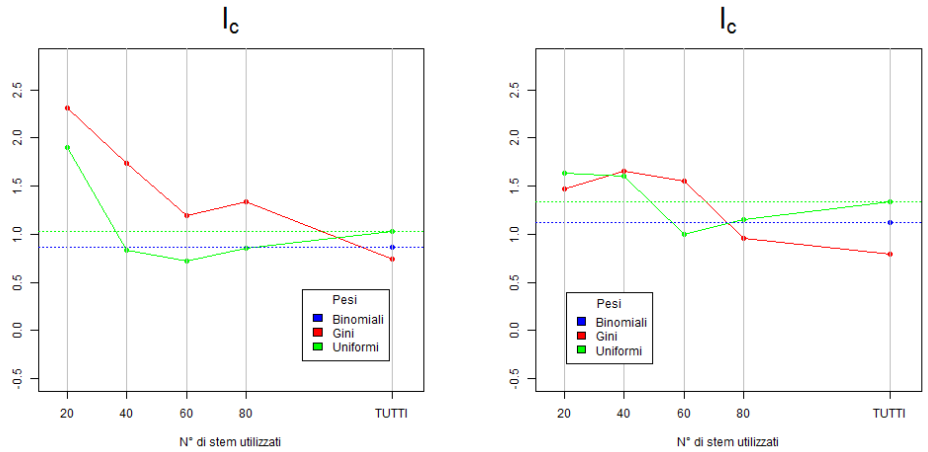
(a) *15 stem per sottoinsieme.* ***(b)*** *8 stem per sottoinsieme.*

Figura 4.18: *Confronto dell'indice I_d per la sentiment analysis dei testi in inglese. Le linee tratteggiate sono di riferimento poiché sono i risultati che si hanno con il metodo originale.*

medio dell'indice di Gini. Tali valori sono anche inferiori rispetto a quelli ottenuti usando 8 stem per sottoinsieme, dal grafico in figura 4.18b si ha che la strategia precedente è comunque la migliore.

Nella figura 4.19 sono riportati invece i valori di I_c ottenuti confrontando la categoria dei testi positivi con quella dei testi negativi. Si può notare che se si fosse interessati ad avere delle buone stime solo delle categorie positivo/negativo le strategie migliori sarebbero la seconda usando tutti gli stem oppure la terza utilizzando 60 stem. In tutti i casi il valore di I_c è positivo, quindi i rispettivi classificatori tenderanno a sovrastimare la categoria dei negativi e sottostimare quella dei positivi.
Si tenga sempre presente che a causa della bassa numerosità dei tweet nella categoria *negativo* la valutazione dell'errore non è detto sia precisissima.
Si osserva comunque che la strategia che porta ad un valore più vicino allo 0 è la terza utilizzando 60 stem e 15 stem per sottoinsieme nel bagging.

Strategie ottimali. Quindi sembrerebbe che in questo dataset per avere delle buone stime si debbano selezionare i 60 stem che hanno un potere predittivo maggiore tramite il decremento medio dell'indice di Gini e poi attribuirgli dei pesi uniformi, usando nella fase di bagging 15 stem per sottoinsieme.
Come si può vedere dalla figura 4.20 gli intervalli di confidenza contengono in tutti i casi il valore vero, questo garantirà delle stime ottimali.



(a) *15 stem per sottoinsieme.* ***(b)*** *8 stem per sottoinsieme.*

Figura 4.19: *Confronto dell'indice I_c per la sentiment analysis dei testi in inglese. Le linee tratteggiate sono di riferimento poiché sono i risultati che si hanno con il metodo originale.*

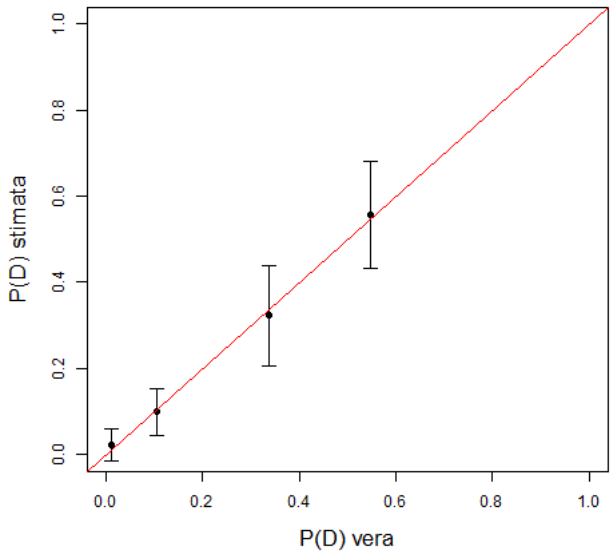


Figura 4.20: *Confronto tra le percentuali vere e quelle stimate di P(D) e $IC_{0.95}$ nel testing set. Sentiment analysis dei testi in inglese.*

4.3.4 Applicazione dei sinonimi

A questo punto si vorrebbe applicare la sostituzione tramite i sinonimi spiegata nel capitolo 3. Come detto lo scopo è quello di ridurre il numero di stem, e quindi la sparsità della matrice $S_{i,j}$, e inoltre quello di permettere un riconoscimento da parte del classificatore di tutti quegli word stem profiles che esprimono lo stesso concetto ma con parole differenti.

Per fare questo lavoro di sostituzione di ciascuna parola con il rappresentante della classe è stato utilizzato un vocabolario creato all'università di Princeton, denominato WordNet, che racchiude al suo interno una sezione dedicata ai sinonimi. Per poter accedere a tale vocabolario è stato usato un pacchetto R (chiamato appunto *wordnet*, Feinerer (2015)) che è in grado di interfacciarsi con il vocabolario e che, data una parola, con una semplice funzione restituisce tutti i suoi sinonimi. In particolare è stata scaricata la versione WordNet 3.1 del vocabolario.

Caratteristiche del pacchetto. Considerando il fatto che avere a che fare con il linguaggio è molto complicato, avere a disposizione una funzione di questo genere è un grandissimo vantaggio, ma lo si è dovuto adattare allo scopo della tesi. Questo pacchetto lavora in lingua inglese. Ci sono dei dizionari anche per tante altre lingue, ma in questo momento non esiste un modo per avere un'interfaccia con il software *R*.

In base a quello che si vuole fare ha comunque dei difetti, per esempio nel momento in cui si cercano i sinonimi di una parola alcuni di essi hanno a fianco una nota rappresentata da una lettera racchiusa tra parentesi tonde. Il problema è che non si conoscono tutte le possibili note che sono presenti e quindi le possibili lettere (o altre diciture varie), inoltre non si sa da quante parole è composto ciascun sinonimo (alcuni sostantivi infatti hanno dei sinonimi composti da più parole). Questo problema è stato risolto facendo la scelta di prendere in considerazione solo la prima parola, quindi ciascuna classe di equivalenza è stata creata solo ed esclusivamente dalla prima parola di tutti i sinonimi.

Un altro aspetto critico con cui ci si è scontrati è dovuto al fatto che nel momento in cui si richiede l'elenco dei sinonimi di un termine bisogna specificare il tipo di sinonimo desiderato. Questo è logico in tutte le lingue, per certe parole non esiste un solo tipo di sinonimi, ma ce ne possono essere vari in base a come le si usano. Le tipologie presenti nel pacchetto sono: *nome*, *verbo*, *avverbio* e *aggettivo*. Come è stato spiegato questa suddivisione è dovuta al fatto che un termine ha vari significati e in base ad essi potrebbe avere dei sinonimi differenti. Per questo è stata resa necessaria la decisione di eseguire delle sostituzioni solo nel caso in cui ci sia un solo *tipo* di sinonimi,

negli altri casi tutto rimane invariato.
Questo aspetto mette in luce anche il fatto che non vale la proprietà transitiva: se *parola1* è sinonimo di *parola2* e *parola2* è sinonimo di *parola3*, allora non è detto che *parola1* e *parola3* siano tra loro sinonimi. Ma non ci si preoccupa di questo aspetto perchè se si dovesse verificare quanto detto allora si avrebbero più gruppi di sinonimi e quindi nessuna sostituzione verrà effettuata.
Le difficoltà di applicazione del metodo sono evidenti, ma in realtà potrebbe portare comunque a dei miglioramenti. Basterebbe infatti la sostituzione di pochi stem per avere degli effetti, purché tali stem siano significativi.

Si è proceduto quindi alla sostituzione dei sinonimi in tutto il dataset in inglese. A causa di alcuni problemi di cattiva codifica, che ci sono stati in generale su tutti i dataset e di cui si discuterà in modo più preciso e approfondito successivamente, e per poter applicare tutte le funzioni che consentissero il funzionamento della sostituzione automatica dei sinonimi, si è provveduto ad eliminare manualmente alcuni caratteri che non venivano riconosciuti da alcune funzioni. Questo ha portato una serie di problematiche, relative alla codifica, e alla variazione di alcuni caratteri che hanno reso più difficile lavorare con il testo.

Si è proceduto comunque ad analizzare per lo meno le situazioni pù semplici. Si è deciso di andare avanti poiché si è notato che i cambiamenti non sono su delle parole a caso, ma se una parola veniva cambiata, tutte le sue occorrenze in tutti i testi cambiavano nello stesso modo, quindi teoricamente la parola è rimasta comunque identificabile rispetto a tutte le altre.

Per poter eseguire le sostituzioni sono stati usati vari pacchetti che trattano variabili di tipo stringa come *stringi* (Gagolewski and Tartanus, 2014), *stringr* (Wickham, 2012), *TextWiller* (Solari et al., 2013) e il già citato *tm*.

Nella tabella 4.13 si riportano alcune informazioni ricavate dalla fase di sostituzione. Infatti sono state create delle funzioni che realizzano un vocabolario dei sinonimi (ristretto alle parole presenti nel dataset) nello stesso momento in cui venivano eseguite le sostituzioni. Questo è completamente accessibile e l'esperto potrà analizzare a posteriori le sostituzioni fatte e avere il controllo sulle parole presenti nel dataset.

Tabella 4.13: *Sinonimi in numeri.*

Numero di classi di sinonimi	267
Numero di parole singole (non sostituite)	12 811
Numero medio di parole per classe	1.84
Numero di parole coinvolte nella sostituzione	464

Come si può osservare il numero di parole coinvolte nella sostituzione non è molto alto, si ricorda però che tra le parole singole che non sono state sostituite ci sono anche tutte quelle strane, che hanno errori grammaticali, che sono incomplete o troncate. Nonostante la bassa numerosità, se si pensa che anche solo alcune di quelle coinvolte siano tra le 129 estratte dal training set e usate per la costruzione del classificatore, l'impatto potrebbe essere significativo.

Nella tabella 4.14 sono riportati i valori dei soliti indici usando i testi dopo la sostituzione, mentre nella tabella 4.15 si riportano nuovamente i risultati analoghi prima di effettuare la sostituzione.

Tabella 4.14: *Sinonimi, sentiment analysis dei testi in inglese. Valori degli indici I_d e I_c al variare del numero di stem per sottoinsieme.*

		15 stem per sottoinsieme		**8 stem per sottoinsieme**	
	N°stem	I_d	I_c	I_d	I_c
Strategia 1	T	0.018	1.162	0.028	1.550
Strategia 2	T	0.030	0.741	0.038	0.758
Strategia 3	T	0.022	1.363	0.032	1.302

Tabella 4.15: *Sentiment analysis dei testi in inglese prima della sostituzione. Valori degli indici I_d e I_c al variare del numero di stem per sottoinsieme.*

		15 stem per sottoinsieme		**8 stem per sottoinsieme**	
	N°stem	I_d	I_c	I_d	I_c
Strategia 1	T	0.021	0.866	0.017	1.126
Strategia 2	T	0.044	0.742	0.078	0.797
Strategia 3	T	0.012	1.030	0.035	1.339

Si può osservare che il valore più basso lo si ha con la strategia già esistente con i pesi binomiali e 15 stem per sottoinsieme. Ma la cosa molto interessante è che questo valore risulta inferiore rispetto al suo equivalente senza l'uso dei sinonimi. Infatti in questo caso si ha $I_d = 0.018$, mentre in precedenza era stato ricavato $I_d = 0.021$. Al contrario invece si osserva che nella strategia 3 la situazione è esattamente opposta, in questo caso l'indice è aumentato. Per quanto riguarda l'indice I_c invece il suo valore risulta sempre positivo, quindi si tenderà a sottostimare i positivi e sovrastimare i negativi, ma è leggermente aumentato.

In seguito vedremo brevemente quali sono state le percentuali stimate con la prima strategia.

4.3.5 Analisi dei risultati

Si riportano ora i risultati delle stime fatte usando la strategia 3 con 60 stem, che risultava la migliore nella sentiment analysis sul dataset in lingua inglese. Dalla figura 4.21 si osserva che anche stavolta la maggior parte dei tweet è

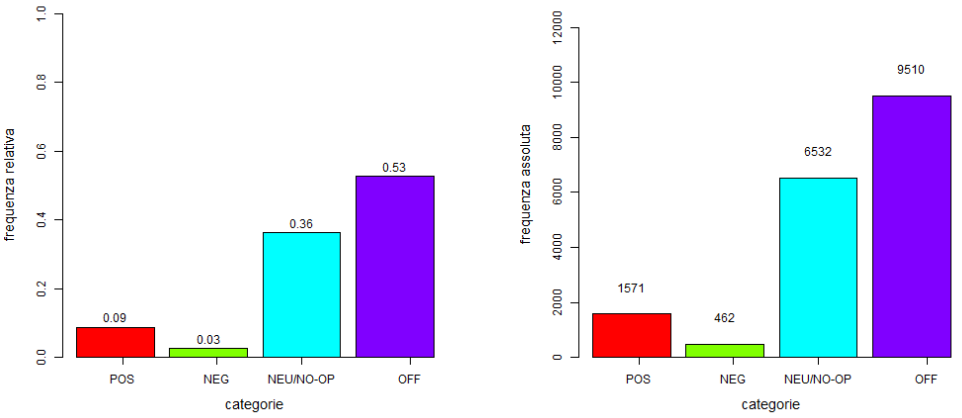


Figura 4.21: *Stime delle percentuali e dei valori assoluti di tweet in ciascuna categoria di sentiment dei testi in inglese.*

stimata come off-topic e no-opinion, quindi anche per la popolazione in lingua inglese il tema Expo ancora non ha suscitato un grande interesse, anzi è sentito ancora di meno visto il minor numero di tweet. Inoltre la percentuale attribuita al sentiment positivo è maggiore rispetto a quella attribuita al sentiment negativo, infatti il numero di tweet positivi è tre volte superiore a quello dei tweet negativi.

Analisi temporale. Analogamente a quanto fatto sui testi in italiano, si è proceduto ad eseguire delle stime per le categorie della sentiment analysis ad intervalli di 5 giorni. In questo caso si hanno 6 registrazioni poiché la finestra complessiva è più ampia.

Come si può osservare dalla figura 4.22 la percentuale di testi negativi è rimasta pressoché costante su valori bassi attorno al 3%, mentre c'è una graduale crescita del sentiment positivo senza nessun particolare picco. Si nota invece un chiaro scambio di massa tra le ultime due categorie (off-topic e neutro/no-opinion).

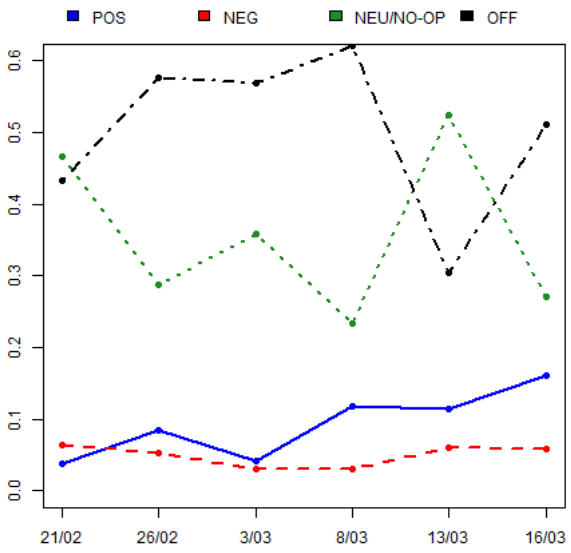


Figura 4.22: *Analisi temporale del sentiment dei testi in inglese.*

Cenno ai risultati sui sinonimi. Giusto per completezza, ma soprattutto per curiosità, si riportano anche i risultati ottenuti con la strategia 1 applicando la sostituzione dei termini sinonimi.

Dalla figura 4.23 si osserva che le stime ottenute sono praticamente uguali. Viste le problematiche che ci sono state e che saranno chiarite e descritte in dettaglio anche più avanti, si può dire che questo approccio andrebbe investigato in condizioni migliori, ma per ora il primo impatto è stato abbastanza positivo, per lo meno i risultati ottenuti seguono lo stesso andamento dei precedenti. Bisognerebbe quindi approfondire la questione, risolvere alcuni problemi e testare la procedura per vedere se le cose possono migliorare.
Un aspetto che sicuramente andrebbe approfondito è quello riguardante il tipo di sostituzione che si può fare. Si ricorda infatti che si è deciso di eseguire la sostituzione solo nel caso in cui il termine appartenga ad un solo tipo di sinonimi (o solo *nome*, o solo *aggettivo*, ecc.), ma potrebbero essere fatte altre scelte più opportune che possano portare ad ulteriori miglioramenti.

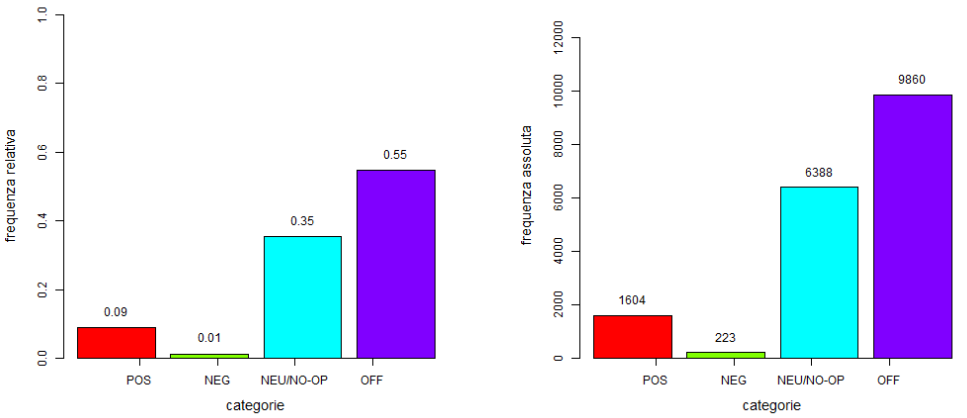


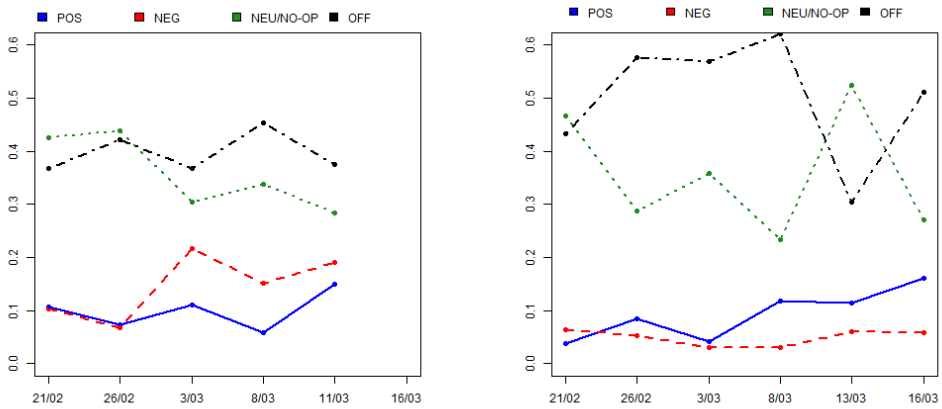
Figura 4.23: *Sinonimi. Stime delle percentuali e dei valori assoluti di tweet in ciascuna categoria di sentiment dei testi in inglese sui quali è stata fatta la sostituzione.*

4.4 Ricapitolazione e commenti

Alla luce di tutte le analisi fatte si può dire che si notano due tendenze opposte tra il popolo di lingua italiana e quello di lingua inglese (si osservi il confronto nella figura 4.24).

Nel caso dei testi in inglese la percentuale dei testi positivi risulta tre volte superiore rispetto a quelli negativi, mentre nei testi italiani accade l'opposto, i tweet negativi sono quasi il doppio di quelli positivi. I motivi potrebbero essere molteplici, ma principalmente si potrebbe dire che questo risultato è dovuto al semplice fatto che l'evento non è ancora iniziato e tutti gli scandali, le inchieste, i problemi sui ritardi e tante altre questioni di questo genere hanno avuto e stanno avendo un forte eco in Italia, ma non all'estero. Questo potrebbe spiegare la ridotta presenza di tweet negativi, che ricordiamo ha creato anche dei problemi nella costruzione del classificatore nel momento in cui si assegnavano le etichette ai testi.

Dai risultati dei tweet in lingua italiana però si ricava un aspetto che, se gli organizzatori lo ritenessero importante, potrebbe essere migliorato. Infatti se la polemica del maialino sardo prendesse piede e questa scelta fatta andrà a ledere la reputazione o l'immagine di Expo (anche perché dall'altra parte sono stati concessi dei permessi per la vendita di insetti da parte di ristoratori asiatici) essi potrebbero decidere di concederne il consumo. Sia chiaro questo è un esempio, forse abbastanza banale e che ha poco impatto, ma la sentiment e l'opinion analysis servono proprio a capire quali sono e di



(a) Tweet in italiano. *(b) Tweet in inglese.*

Figura 4.24: *Confronto fra i risultati della sentiment analysis dei tweet in italiano ed in inglese.*

che portata sono i fenomeni come questo, che potrebbero essere presi sotto gamba dagli organizzatori.
Nell'intervallo di tempo analizzato non è stato ricavato invece nessun particolare aspetto di cattiva gestione.
Si tenga ben presente che i risultati ottenuti (o che si otterranno in futuro, anche una volta risolti tutte le problematiche avute) vanno ponderati ed analizzati alla luce del fatto che la maggior parte dei testi sono retweet, quindi si devono tenere sotto controllo eventuali fenomeni di *sindrome da follower*, ovvero quando i fan o ammiratori dei personaggi influenti o famosi ritwittano qualsiasi cosa essi pubblicano, senza per forza essere d'accordo su quanto scritto. Un altro aspetto è invece legato alla grande presenza dei link esterni, questo causa una frammentazione dell'informazione. In tal caso infatti non sempre si è certi di quale sia l'opinione dell'utente poiché parte dell'informazione sta nel link esterno. Un lavoro che si potrebbe fare è quello di decodificare gli *url* o i *mini-url* e vedere se ce ne sono alcuni che hanno un'ampia diffusione e che quindi trasportano con se il sentiment che le persone vogliono esprimere. Sicuramente è molto dispendioso, ma garantirebbe un'analisi più completa.

Si può notare che per ora è possibile fare solo un confronto parziale tra i due dataset, visto che manca l'opinion analysis dei testi in inglese e inoltre anche la sentiment analysis andrebbe fatta in modo più preciso e rigoroso. In particolare si riuscirà sicuramente a risolvere i problemi legati alla scarsa numerosità quando l'evento inizierà o comunque nel periodo subito antecedente l'inaugurazione. Arrivati a quel punto si avranno anche dei dataset

molto più corposi e riguardanti un intervallo temporale molto più ampio che consentirà di mettere in luce i cambi di sentiment che saranno sicuramente molto più frequenti. Si potrebbe anche prendere in considerazione di cambiare gli intervalli di valutazione del sentiment, per ora fissato a 5 giorni, e ridurlo o ampliarlo in base al susseguirsi degli eventi. Analisi del sentiment come quelle appena fatte diventano infatti molto più interessanti e utili se si individuano degli eventi reputazionali, che quindi vengono analizzati tramite i social per capirne e quantificarne l'impatto. Sarebbe quindi interessante individuare questi eventi man mano che si susseguono ed analizzarli a posteriori sui social network.

Per raggiungere un livello di completezza maggiore sarebbe opportuno analizzare vari tipi di dataset, non solo quelli ricavati da Twitter. Si è cercato di scaricare i commenti fatti sui post pubblicati dalla pagina ufficiale dell'evento su Facebook usando l'applicazione *netvizz* (Rieder, 2013), ma ci sono stati dei problemi sulla lettura dei dati che verranno spiegati nell'ultima parte. Quindi sarebbe interessante, ma soprattutto più completo, fare un'analisi di questo genere più ampia, soprattutto perché il popolo di Twitter e quello di Facebook possono essere molto diversi tra loro.

Un altro campo su cui si potrebbe applicare la sentiment e opinion analysis per studiare i pareri su Expo è quella delle testate giornalistiche, che darebbero un'analisi da un punto di vista più istituzionale, ma anche di blog o qualsiasi tipo di pagina online a cui si possa avere accesso e che potrebbe influenzare l'opinione pubblica, analizzando se questa contaminazione è possibile.

Si rimanda all'appendice B per un'analisi dei pesi che sono stati usati nelle diverse strategie.

Conclusioni e sviluppi futuri

L'obiettivo di questo lavoro di tesi è stato quello di analizzare i social network tramite la sentiment analysis. Sono stati presentati tutti i metodi esistenti tra i quali è stato scelto quello che si adatta meglio allo scopo prefissato, ovvero il metodo di classificazione aggregata di Hopkins e King. Questo modello è stato descritto nei minimi particolari e sono stati presentati i suoi punti di forza e i suoi limiti.
Successivamente sono state descritte due strategie alternative a quelle presentate dai due autori che hanno lo scopo di migliorare le prestazioni del metodo. La prima novità è quella di usare la classificazione individuale, precedentemente scartata in favore di quella aggregata, per selezionare e assegnare i pesi alle parole. In particolare è stato usato il decremento medio dell'indice di Gini, che viene calcolato dall'algoritmo Random Forest per la costruzione degli alberi, per stilare una classifica degli stem partendo da quello che ha potere predittivo maggiore. Quindi è stato sfruttato l'ordine della classifica per applicare il metodo con un sottoinsieme di stem e allo stesso tempo è stato assegnato agli stem un peso proporzionale all'incremento medio dell'indice di Gini.
La seconda novità invece è quella di eseguire fin dall'inizio una sostituzione delle parole tramite l'uso dei sinonimi, in particolare sono state create delle regole decisionali (necessarie vista la complessità del linguaggio) che permettano di semplificare il testo. Infatti la sostituzione di una serie di termini sinonimi tra loro con un unico rappresentante riduce il numero di stem presenti nel corpus e quindi porta ad una riduzione dimensionale.

Il metodo di Hopkins e King e le nuove strategie sono state applicati ai tweet sull'Expo 2015 di Milano. In particolare sono stati scaricati i tweet in lingua italiana e i tweet in lingua inglese. I tweet in italiano sono stati analizzati sia dal punto di vista della sentiment analysis che dal punto di vista dell'opinion analysis, mentre sui tweet in inglese è stata eseguita solo la sentiment analysis a causa della scarsa numerosità dei testi nelle varie categorie. Solo in quest'ultimo caso è stato possibile applicare la sostituzione tramite i sinonimi. In entrambi i casi sono stati forniti sia dei risultati globali relativi al periodo di analisi, sia delle analisi temporali che descrivono l'andamento del sentiment istante per istante.

Dalle analisi dei tweet in italiano sono state scelte due strategie ottimali, una per la sentiment e una per l'opinion analysis. Nel primo caso la strategia ottimale è quella dei due autori, usando quindi dei pesi binomiali, mentre nel caso dell'opinion analysis la strategia ottimale è quella di costruire il modello selezionando i primi 80 stem dalla classifica e pesarli con il decremento medio dell'indice di Gini.
Per quanto riguarda i tweet in inglese la strategia ottimale è quella di selezionare invece i primi 60 stem usando sempre i pesi proporzionali al decremento dell'indice di Gini. Applicando la sostituzione tramite i sinonimi si ottiene che in alcune strategie si ha un miglioramento delle prestazioni del classificatore, ma a causa di alcuni problemi di codifica del testo si preferisce rimandare ad una valutazione migliore di questa strategia.

Analizzando quindi i risultati si osserva che sia per quelli in italiano sia per quelli in inglese le categorie che hanno proporzioni maggiori sono quelle di testi off-topic e no-opinion, mentre le caratteristiche del vero e proprio sentiment sono differenti nei due casi. Dai testi in italiano si osserva che prevali il sentiment negativo (doppio rispetto a quello positivo) e in particolare si osserva una predominanza di opinioni negative generiche legate all'evento e alle scelte fatte dagli organizzatori. Dall'analisi delle serie temporali si osserva che il sentiment negativo è legato ad una protesta ben precisa legata al divieto di consumare il maialino sardo durante il periodo dell'esposizione, fenomeno che nasce, raggiunge picchi del 15% del sentiment negativo e si esaurisce nell'intervallo di tempo studiato.
Per i testi in inglese le cose sono esattamente opposte, ovvero c'è una predominanza del sentiment positivo rispetto a quello negativo (con un rapporto di 3 a 1), dall'analisi temporale però non emergono delle grosse variazioni o fenomeni simili a quello osservato nei testi in italiano.

Complessivamente si può quindi dire che le novità presentate hanno portato dei buoni risultati, riuscendo ad eguagliare e in alcuni casi migliorare le prestazioni del metodo con le strategie preesistenti. Questo vuol dire che si dovrebbe continuare nella ricerca di nuove metodologie per pesare e scegliere in modo opportuno le parole oppure per cercare di ridurre la complessità numerica che caratterizza il metodo senza eseguire il bagging.

A questo punto si vogliono descrivere le problematiche avute e i possibili sviluppi futuri ad esso legate.

Il primo problema che si vuole presentare non è di carattere tecnico, ma rappresenta un punto chiave per fare una buona analisi del sentiment tramite il metodo non parametrico di Hopkins e King, questo è l'hand coding. Questa

procedura deve essere eseguita da più persone esperte nel settore del problema che si sta affrontando, bisogna essere molto precisi, accurati e attenti. Come spesso è stato detto, il risultato finale è strettamente legato alla buona riuscita di questa procedura. Si ricorda che le caratteristiche fondamentali di un buon hand coded set sono la buona rappresentatività linguistica di ciascuna categoria e anche il rispetto dell'assunzione (2.7) ($\mathbf{P^h}(\mathbf{S}|\mathbf{D}) = \mathbf{P}(\mathbf{S}|\mathbf{D})$), ovvero i testi etichettati a mano devono essere tali da rappresentare le stesse costruzioni linguistiche di tutti i testi di tutte le categorie. Quindi teoricamente non bisogna guardare solo le numerosità e quindi verificare di avere un buon numero per ogni categoria, ma anche verificare che ciascuna di esse sia rappresentata da testi con una certa variabilità linguistica che consenta di catturare tutti i possibili testi che esprimono lo stesso sentiment (o genericamente che appartengono alla stessa categoria) presenti in tutto il corpus. Si ribadisce che non è necessaria la presenza di un campione statisticamente rappresentativo.
Nel caso che è stato affrontato, ancora una volta, è stato fatto quanto di meglio era possibile fare. Il tagging infatti è stato fatto da un'unica persona e anche avendo parlato e discusso sulle caratteristiche delle diverse categorie con la dottoressa Arena, si è proceduto più che altro cercando di fare delle scelte logiche e basate sul buon senso e su quello che sembrava più ragionevole fare. Non si nascondono evidenti problemi e indecisioni avute durante questa fase che ha portato sicuramente a delle assegnazioni non proprio adeguate e delle volte neanche coerenti.
Complice della nascita di questo problema è sicuramente il tipo di dataset, o meglio il tema trattato. A parte in alcuni casi palesi, non è stato facile capire se un tweet fosse off-topic oppure no, infatti non sempre si era a conoscenza di tutti gli eventi strettamente legati ad Expo oppure se un evento è patrocinato da Expo oppure ancora se l'evento è completamente estraneo ed è stato usato uno degli hashtag usati per la ricerca solo per avere popolarità in rete. Delle volte era difficile distinguere anche un tweet positivo da uno no-opinion alla luce della regola stabilita di considerare positivi i retweet di tweet positivi, anche se scritti da utenti istituzionali. Un testo come *"Fantastica visuale dall'alto dei padiglioni!"* scritta dall'account ufficiale di Expo e poi retwittata da un altro utente per alcuni potrebbe essere positivo per altri invece un semplice tweet senza opinione. Quindi bisogna tenere presente che nei casi in cui il tagging è fatto da persone non esperte, potrebbe entrare in gioco un fattore soggettivo che potrebbe distorcere i risultati finali.
Per quanto riguarda il caso di Expo affrontato si è cercato comunque di essere coerenti il più possibile, è stato infatti riportata una descrizione dettagliata di ogni categoria con le scelte che sono state fatte proprio durante la fase di tagging.
Per risolvere questo problema, avendo l'obiettivo di verificare se le tecniche e le integrazioni al metodo che sono state presentate in questa tesi hanno un vero e proprio impatto positivo sul tema della classificazione testuale e della

sentiment analysis, le si potrebbe applicare ad una dataset adatto a testare i classificatori di questo tipo. Nel lavoro presentato è stato cercato un problema legato ad un tema interessante e attuale come l'Expo, ma nell'ottica di testare un classificatore si dovrebbe scegliere un problema in cui fare il tagging manuale sia una pratica semplice e quindi le categorie siano di facile individuazione e riconoscimento. Per esempio lavorare su dati relativi a delle elezioni politiche o comunque un tema in cui gli utenti tendono ad appoggiare in modo chiaro un'idea piuttosto che tante altre. Questo non vuol dire che non siano ammessi casi intermedi, ma questi sono a loro volta facili da riconoscere. Tutte queste scelte permetterebbero un tagging più agevolato, ma allo stesso tempo più preciso che permetterà di fare delle analisi conclusive mirate sulla metodologia usata al netto degli errori o delle scelte fatte durante la fase di etichettatura.

Una cosa che non è stata fatta per mancanza di tempo è l'analisi di robustezza del metodo provando a cambiare per esempio il numero di testi che sono stati taggati, vedendo se aumentando la numerosità il metodo diventa sempre più preciso. Questo aspetto è importante soprattutto se si stanno trattando dati scaricati dai social network, infatti le numerosità a cui si fa riferimento per ora sono quelle presentate da Hopkins e King ma su dati completamente differenti. Si ritiene infatti che la numerosità sia strettamente dipendente dal tema trattato: il caso di Expo ha messo in luce che più del 50% delle volte gli hashtag che fanno riferimento all'esposizione sono usati in modo improprio, di conseguenza questa questione andrebbe analizzata con cura una volta che è stata fatta un'analisi iniziale ed esplorativa del dataset che si andrà ad utilizzare.
Si potrebbe inoltre testare la robustezza del metodo, soprattutto la parte relativa alla selezione degli stem e assegnazione dei pesi tramite l'indice di Gini, studiando se c'è una relazione tra il numero di stem usato in ciascun sottogruppo nella fase di bagging con altri fattori tra cui il numero e il tipo di categorie, il tipo di dataset e la lingua usata nei testi. Per ora infatti si è solo provato a ridurre gli stem da 15 a 8 per sottoinsieme, ragionando sul fatto che il testo dei tweet è corto, ma si è osservato che questa non può essere la ragione della scelta. Infatti in alcuni casi usandone 15 si ottenevano dei valori degli indici migliori rispetto ad usarne 8.

Per quanto riguarda la parte relativa alla sostituzione con i sinonimi andrebbe testata nuovamente una volta risolti i problemi di codifica, ma andrebbero riviste sicuramente altre cose tra cui ridurre il costo computazionale richiesto per individuare ciascuna parola, costruire la classe di sinonimi ed eseguire eventualmente la sostituzione. Si dovrà lavorare soprattutto per snellire il codice che è stato creato per ora, facendo delle scelte più adatte a trattare le parole come variabili di studio. Inoltre si potrebbe continuare a cercare la presenza di vocabolari più adatti, oppure altri software che facilitino l'uso

di questo approccio.

Il secondo problema che si vuole presentare è tipico in campi di ricerca recenti e in continuo sviluppo ed è la mancanza di pacchetti adatti a trattare problemi di sentiment analysis che siano stabili e robusti. Qui si parla principalmente di quelli che sono presenti nel software R e vengono continuamente aggiornati, presentando sempre delle nuove funzionalità. Il fatto è che ogni volta che avvengono questi aggiornamenti alcune delle vecchie funzionalità non si è in grado di usarle.
Così è successo con il pacchetto *TwitteR*, a metà febbraio 2015 è stata rilasciata l'ultima versione che presentava la nuova funzione *search_ twitter_ and _ store* che, come già detto, consente uno stoccaggio veloce e sicuro di tutti i tweet recenti mancanti al dataset. Il problema è che ora la funzione non consente di scaricare i tweet in base alla geo-localizzazione, di conseguenza l'idea iniziale di scaricare i tweet in base alla provenienza geografica è stata abbandonata e si è optato per la selezione tramite la lingua, che sicuramente risulta essere meno efficace. Inoltre sempre questa stessa funzione ha dei limiti sul numero di tweet scaricabili (valore fissato a 5000), cosa che invece la funzione *search_ twitter*, che ha le stesse funzionalità della precedente ma non procede al salvataggio dei dati su nessun file esterno, non ha. Infatti con l'ultima funzione si possono scaricare tutti i tweet che si desiderano avendo come unici limiti quelli imposti dall'API di Twitter (quindi non più di 18 000 ogni quarto d'ora e tweet non più vecchi di una settimana).
Quindi si è preferito usare l'ultima versione, dando la precedenza ad eseguire un download sicuro dei dati e rinunciando ai tweet geo-localizzati.
In generale non c'è modo di raggirare e risolvere questo problema, si deve essere sempre pronti e aggiornati sui cambiamenti ed eventualmente testare prima le nuove versioni e poi decidere di tenerle oppure continuare ad usare le vecchie. Si osserva quindi che risolvere le questioni legate alla codifica è fondamentale anche per questo problema, infatti i pacchetti potrebbero essere molto sensibili e dare risultati scorretti nel momento in cui vengono applicati ad un testo mal codificato.

Il vero grande problema che si è presentato fin da subito è legato alla codifica dei testi. Come specificato nella sezione dedicata al download dei dati, è stata utilizzata la funzione *search_ twitter_ and_ store* del pacchetto *TwitteR*, che in modo automatico aggiorna un file sql, in cui sono presenti i vecchi tweet, con tutti i tweet più recenti. Questa procedura è fondamentale perché se fatta manualmente avrebbe potuto portare a degli errori come perdita di dati o duplicazione degli stessi. Questa funzione però presenta due problemi: il primo è un limite nel numero di dati che possono essere scaricati, infatti esso potrà aggiornare il vecchio file con massimo altri 5000

tweet, quindi è necessario eseguire l'aggiornamento più volte al giorno, anche perché il numero di tweet che vengono inviati giornalmente usando un certo hashtag (quelli sull'Expo, caso che si è studiato) non è costante, ma potrebbe presentare delle forti variazioni in base agli avvenimenti che si susseguono.
Il secondo difetto invece è proprio quello legato alla cattiva codifica. Infatti, una volta che si procede alla lettura del file con tutti i dati, si nota che i testi non vengono codificati bene, alcuni caratteri non sono riconosciuti e vengono sostituiti da altri. Si è cercato in vari modi di risolvere questo problema, usando varie funzioni come *iconv* oppure altre fornite dalla libreria *stringi*, ma non è stato possibile mettere a posto il testo. Si è deciso quindi di continuare a fare tutte le analisi sapendo però che il testo portava con se anche questi tipi di caratteri.
Come è stato spiegato man mano che si presentavano, questo ha creato una serie di problemi nell'applicazione di tutti i metodi che ci si era proposti di analizzare. Si è parlato in precedenza del problema legato al RandomForest nel momento in cui ad ogni stem del training set si deve associare il relativo peso. La funzione *ReadMe*, che si interfaccia anche con altri programmi come Python 2.7, risulta essere più sofisticata e fine riuscendo a riconoscere i caratteri strani ed eliminarli. Quando si applica il Random Forest invece non si è stati in grado di avere lo stesso grado di pulizia del testo, quindi nel momento in cui si dovevano assegnare i pesi agli stem forniti dal pacchetto *ReadMe* non c'era un match con l'elenco degli stem e il relativo decremento medio dell'indice di Gini e di conseguenza a tali stem veniva assegnato un peso nullo.
Questo stesso problema lo si è avuto nell'applicazione dei sinonimi, se le parole infatti presentano problemi di codifica non possono essere riconosciute dal vocabolario e per forza vengono considerate come parole insostituibili. In realtà, non solo si ha poca completezza nella sostituzione a causa di quanto appena detto, ma è stato anche difficile trovare delle funzioni che dessero dei risultati concordanti nel momento in cui si lavorava con dei testi che presentavano parole male codificate. Giusto per fare un esempio, nel pacchetto *stringi* esistono due funzioni, *stri_extract_all_words* tramite il quale si estraggono tutte le parole di un testo e *stri_stats_latex* tramite il quale si conta il numero di parole del testo. Nonostante facciano parte entrambe dello stesso pacchetto, la presenza di caratteri particolari fa si che spesso siano in disaccordo ovvero: se si conta il numero di parole estratte usando la prima funzione, esso differisce (in alcuni casi, si suppone dipenda dai caratteri che creano il problema) dal numero di parole che invece viene conteggiato usando la seconda funzione. Questo problema si è verificato più volte nell'applicazione delle funzioni create per fare la sostituzione tramite i sinonimi, nel tentativo di far girare comunque il codice si è provato ad escludere in modo forzato i caratteri particolari, ma questo, come è stato detto nei capitoli precedenti, ha completamente alterato il testo.
C'è stato inoltre un altro problema legato alla codifica, ma in questo caso

dovuto alla poca esperienza nel trattare testi in due diversi sistemi operativi. Infatti funzioni come quella creata per eseguire la sostituzione dei sinonimi oppure i pacchetti *TwitteR* e analoghi per lo scaricamento dei dati sono stati utilizzati su Windows 7, dove erano stati installati inizialmente. Il pacchetto *ReadMe* invece non si è riusciti a farlo funzionare su Windows, vari utenti hanno avuto lo stesso problema e non è stata ancora trovata una soluzione. Di conseguenza si è dovuto lavorare obbligatoriamente su un altro ambiente, per comodità e semplicità è stato scelto Ubuntu su cui si riesce ad usare il pacchetto *ReadMe*. Vista la poca esperienza si è deciso di lavorare su Ubuntu per fare le stime e di usare Windows per tutte le altre analisi. Purtroppo questa scelta ha determinato la nascita di ulteriori problematiche, mentre si procedeva con il lavoro si è notato infatti che alcuni caratteri scritti in Windows venivano interpretati e codificati male da Ubuntu e viceversa, ne sono un esempio i caratteri accentati. Anche questo può aver contribuito ad una perdita di informazioni ed efficienza dei metodi che sono stati analizzati. Non si mette in dubbio che questi non siano problemi risolvibili, può darsi anche in modo semplice.

Quindi risolvere questi problemi di codifica diventa fondamentale per riuscire a lavorare tranquillamente e concentrarsi sulla metodologia. Questo infatti impedisce di fare un lavoro pulito e su cui si possa fare affidamento. Bisognerebbe capire se gli errori nel codificare il testo sono legati alle funzioni che sono state usate per il download oppure se si generano in automatico poiché potrebbe dipendere da quali caratteri e tramite quale dispositivo ciascun utente invia il testo. Nel primo caso però si dovrebbe indagare se tali funzioni possono essere impostate diversamente o se comunque esiste un modo per eseguire una lettura corretta del testo.

L'ultimo problema invece riguarda il software *netvizz* che si sarebbe voluto usare per scaricare i commenti dei post della pagina Facebook di Expo. Una volta fatta la ricerca, il materiale scaricato si trova nel formato *.tab*, è stato normalmente aperto per dare uno sguardo con LibreOffice, ma una volta importati i dati su R tramite la funzione *read.csv* si è notato che non tutte le righe venivano effettivamente lette. Inoltre anche il file *.tab* presentava delle incongruenze rispetto ai dati dichiarati dal software, che in fase di download da alcune informazioni sul materiale che si sta scaricando, tra cui il numero totale di commenti. Eseguendo un'accurata ricerca sul foglio di calcolo è stato notato che centinaia di celle (quindi di commenti) erano condensate in un unica cella. È stato provato al recupero di questi commenti ma ogni tentativo è stato vano. In base a come si è presentato il dataset si ha l'impressione che i dati non vengano scaricati tutti nello stesso modo, ma è come se alcuni blocchi siano stati salvati usando come separatore il *tab* altri invece la virgola. Nel momento in cui si importano i dati in R oppure in

LibreOffice si seleziona un unico separatore, di conseguenza alcuni blocchi si condensano creando problemi di lettura delle celle nelle quali sono salvati.
Se si procedesse, poiché i blocchi mancanti sono relativi a periodi di tempo continuativi, i risultati sarebbero per forza distorti. Bisognerebbe indagare a fondo e capire se c'è un modo per riuscire a recuperare questa informazione e sfruttare quindi anche questi dati per avere un quadro completo del sentiment della popolazione. Infatti se l'obiettivo è quello di analizzare il sentiment generale della popolazione, non ci si può attenere solo ed esclusivamente ad un unico social network, ma si dovrebbe fare un'analisi più allargata. Come è stato detto sopra infatti ogni social è caratterizzato da utenti differenti, che hanno un modo di esprimersi differente. Questo fattore dipende anche dalla nazionalità della popolazione. Allora sarebbe importante trovare un modo per scaricare i dati da varie fonti. Un primo passo si potrebbe fare risolvendo i problemi con *netvizz*, altrimenti la strada alternativa sarebbe quella di trovare un'azienda che sia disposta ad investire in questo campo, mettendo a disposizione i dati che la riguardano. Quest'ultima risulta una strada molto ardua, infatti le aziende sono molto restie a mettere in mano a degli sconosciuti questo tipo di dati, che stanno diventando sempre più preziosi proprio perché possono dare delle informazioni fondamentali per lo sviluppo di un buon prodotto.

L'ultimo consiglio invece è quello di sfruttare l'unica fonte di dati che si ha per ora, ovvero il pacchetto *TwitteR*. Per poter avere in futuro dei dataset completi ed adatti a fare questo tipo di analisi, bisogna fin da subito pensare a dei temi interessanti e scaricare quanti più tweet è possibile. Per evitare perdita di dati dovuto all'eccessiva richiesta, ricordando che ci sono dei limiti, si dovrebbero fare delle scelte in base al periodo. Per essere chiari se si sta provvedendo al download dei dati in vari dataset ma si sa che in certi giorni alcune ricerche potrebbero fornire una numerosità molto maggiore di tweet, allora si deve dare precedenza a questi download e rimandare di qualche giorno gli altri. Visti i vincoli che sono presenti, bisogna stare molto attenti e continuare ad aggiornarsi in attesa di novità che consentano download più sicuri e liberi e che potrebbero comparire da un momento all'altro.
Si segnala inoltre che solo recentemente è stato trovato un modo per connettere il software *R* con *LinkedIn*, questo potrebbe essere usato per trarre ulteriori informazioni, soprattutto sulle aziende. Non è stato mai provato di conseguenza non si sa esattamente in che modo funziona e cosa consente di fare.

È anche fondamentale continuare ad interagire con i manutentori dei pacchetti fornendo loro consigli ed esponendogli tutte le problematiche che si hanno. Infatti non solo questo contribuirà ad avere in futuro dei prodotti

migliori, ma permetterà uno scambio di opinioni sul tema della sentiment analysis che serve sopratutto a chi, come me, si è approcciato a questo tipo di studi da poco tempo, senza aver mai trattato problematiche di questo genere. Si segnala in particolare Gary King (dell'Harvard University) che ha risposto sempre ad ogni dubbio di natura concettuale sul metodo da lui creato, ma anche pratico sull'uso del pacchetto *ReadMe*.

Appendice A

Come funziona Twitter

Figura A.1: *Screen del profilo Twitter della cantante Anastacia.*

Twitter è un famoso e ormai diffusissimo social network che permette di inviare brevi messaggi, i cosiddetti *tweet*. Il numero totale di caratteri che possono essere inseriti è 140, compresi di punteggiatura e spazi bianchi. Ad accompagnare il testo possono essere inseriti dei link a qualche sito esterno, ma vista la limitazione del numero di caratteri sono stati creati gli *short-url* che sono comunque dei link verso l'esterno, ma molto più brevi e dal quale non è possibile capire verso che sito ci si sta collegando nel caso in cui si decidesse di accedervi. Oltre ai link possono essere inserite delle immagini o dei piccoli video.
Ogni utente è identificato da un nickname che viene preceduto dal simbolo @.

L'utente che si iscrive a Twitter può decidere se rendere il suo profilo pubblico oppure privato. Pubblico vuol dire che chiunque potrà leggere i tweet che scrive, mentre privato vuol dire che solo alcune persone potranno accedervi. Ogni utente ha la possibilità di seguire altri utenti e quindi a sua volta può essere seguito da altri utenti. Si osservi la figura A.1 che rappresenta il profilo ufficiale dell'artista internazionale Anastacia, che usa come nickname *@AnastaciaFanily*. Si indicano con *follower* le persone che la seguono, ovvero che riceveranno nella propria pagina principale tutti i tweet che la cantante invierà, mentre si indicano con *following* le persone che lei segue, per cui lei nella sua pagina principale leggerà ogni tweet che verrà scritto da questo gruppo di persone.

Ovviamente la parte interessante e che rende questa piattaforma un social network è l'interazione tra i vari utenti, che va al di la della lettura dei tweet scritti dagli altri. Questo è possibile tramite alcune funzioni che vengono ora elencate, facendo riferimento alla figura A.2 (in cui la cantante diceva ai suoi fan di essere libera per sempre dal cancro al seno a seguito di una doppia mastectomia):

- **risposta al tweet**, cerchiato in rosso nella figura, consente di rispondere all'utente che ha scritto il post, avendo sempre a disposizione un massimo di 140 caratteri. Il messaggio sarà a quel punto del tipo "@utente *...risposta...*";
- **tweet preferito**, in blu, indica un apprezzamento del tweet che è stato scritto (equivale ad un *mi piace* di Facebook);
- **retweet**, in verde, consente di inoltrare il messaggio aggiungendo, se desiderato e se possibile (se non si sforano i 140 caratteri), altro testo. L'uso di questa funzione è evidente nel testo scritto nella parte bassa della figura precedente (A.1), in cui compare il nuovo testo, le lettere *RT* che indicano l'inoltro del massaggio che segue, che in particolare è stato scritto dall'utente il cui nome è preceduto dal simbolo @.

Hashtag. Di particolare interesse è l'uso dell'hashtag, ovvero il classico cancelletto #, prima di una parola che assume una particolare importanza nel tweet che si vuole inviare. Non solo ha un ruolo di contestualizzare la frase che si è scritto, ma il social network consente ad ogni utente di poter eseguire una ricerca immediata di tutti i tweet (ad esso accessibili) che contengono lo stesso hashtag.
L'uso di questa funzione è comune proprio per far si che tutti gli utenti possano leggere il tweet, commentarlo o ritwittarlo. Ma è usato anche per rendere una discussione (indicata appunto dall'hashtag) virale, facendola diventare quindi un cosiddetto *trend topic*.

Figura A.2: *Esempio di tweet inviato dalla cantante Anastacia.*

Nel caso che è stato studiato (dell'Expo 2015 di Milano) è stato notato che l'hashtag più usato in assoluto è #Expo2015, quindi chiunque fa riferimento all'esposizione universale, oltre alla frase che vuole scrivere, aggiunge quest'hashtag.

Appendice B

Analisi e confronto dei pesi

In questa parte si vuole studiare la relazione che c'è tra i pesi binomiali e quelli proporzionali al decremento medio dell'indice di Gini, per cercare di capire se i pesi assegnati sono diversi tra loro oppure se ci sono dei casi in cui essi concordano.
Si ricorda che i pesi binomiali sono inversamente proporzionali alla varianza della variabile aleatoria associata a ciascuno stem. Sia X_j la variabile aleatoria binaria associata allo stem j, che vale 1 se lo stem è presente e 0 altrimenti, allora essa ha distribuzione Bernoulli di parametro q_j, quindi il peso sarà $p_j = \frac{\frac{1}{q_j(1-q_j)}}{\sum_{j=1}^{J} \frac{1}{q_j(1-q_j)}}$. La stima di q_j è fatta ovviamente in modo empirico sulla base dei testi che si hanno a disposizione. Quindi si assegna un peso maggiore agli stem che compaiono raramente oppure molto spesso, mentre un peso basso ai casi intermedi. Questo sistema di assegnazione non dipende dalle categorie.
Nel caso dei pesi ricavati a partire dall'indice di Gini invece i pesi sono calcolati sulla base della classificazione individuale fatta tramite Random Forest, quindi l'assegnamento è strettamente legato alle categorie.

Tweets in italiano: sentiment analysis. Si riportano i 20 stem che hanno peso maggiore, partendo con il caso della sentiment analysis dei tweet in italiano.

Nella figura B.1 si riporta l'elenco degli stem ai quali è stato associato un peso maggiore rispettivamente con pesi binomiali e tramite l'indice di Gini. Questi poi sono confrontati nella figura B.2 in cui ogni pallino corrisponde ad uno stem, collocato nel grafico in base ai valori dei due indici. Come si può osservare i pesi maggiori vengono assegnati a stem differenti, anzi ten-

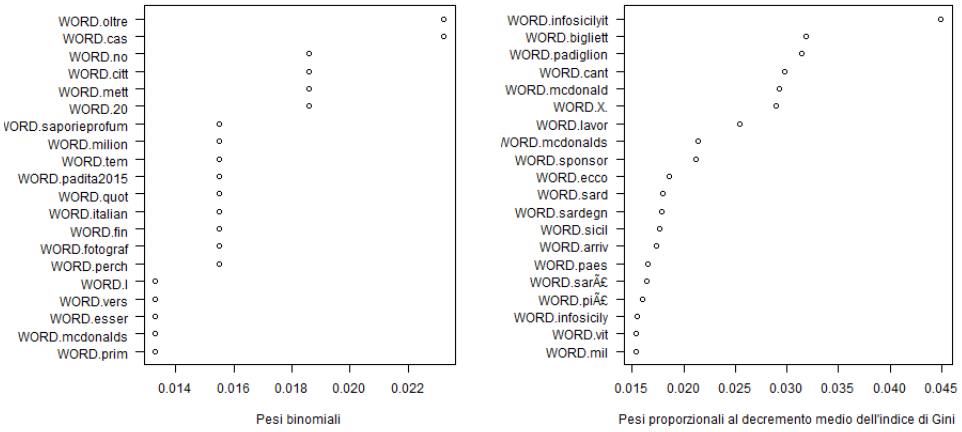


Figura B.1: *Sentiment analysis tweet in italiano. Elenco dei 20 stem con pesi maggiori.*

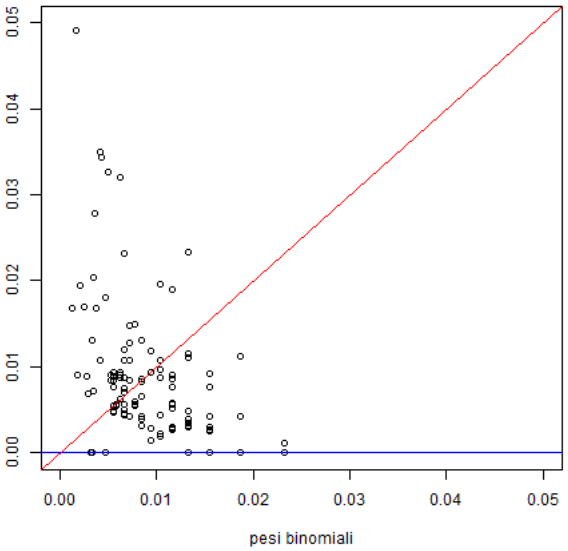


Figura B.2: *Sentiment analysis tweet in italiano. Confronto fra i due sistemi di pesi.*

denzialmente se lo stem ha un peso binomiale alto allora gli è stato associato un peso proporzionale al decremento medio dell'indice di Gini basso, infatti nessun valore alto dei due pesi si trova lungo la bisettrice del quadrante.

Si noti inoltre la presenza di caratteri strani nella denominazione degli stem che non permette quindi di avere un match automatico tra i due pesi (il prefisso *WORD.* deriva dall'output delle funzioni del pacchetto *ReadMe*), per cui nel caso del Random Forest per convenzione era stato assegnato un peso nullo. In questo caso si conta quindi che è stata persa l'informazione di 7 stem che potevano avere un ruolo importante (per esempio *sarà* e *più* avranno peso nullo). Questo mette in luce le difficoltà avute nel trattare i problemi di codifica.

Tweets in italiano: opinion analysis. Si riportano ora gli stem con peso maggiore nel caso dell'opinion analysis. Si noti che i pesi binomiali sono gli stessi del caso precedente.

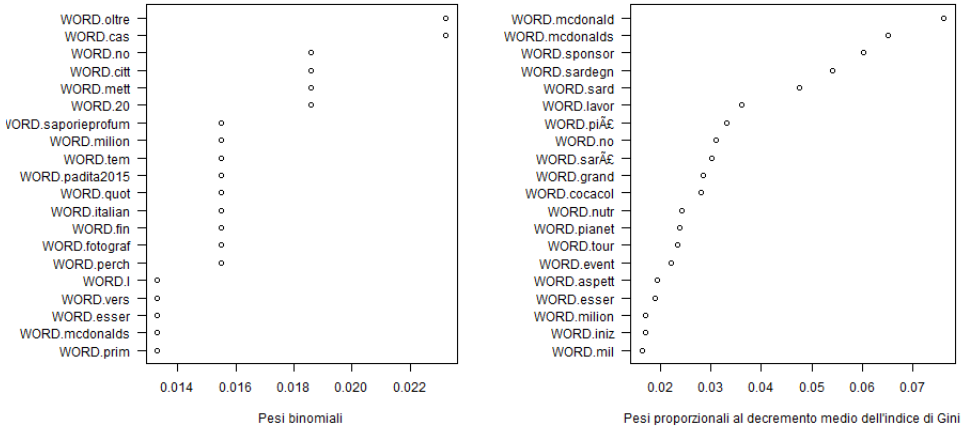


Figura B.3: *Opinion analysis tweet in italiano. Elenco dei 20 stem con pesi maggiori.*

Come si osserva dalle figure B.3 e B.4 la situazione migliora leggermente, ma l'andamento generale è esattamente lo stesso del caso precedente. I due sistemi di pesi sono differenti tra loro.

Tweets in inglese: sentiment analysis. In questo caso cambiano completamente i testi, ma le cose non cambiano neanche in questa situazione, si vedano le figure B.5 e B.6.

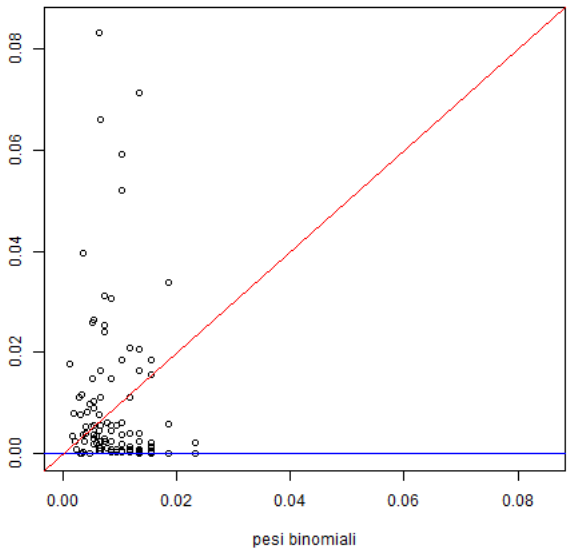


Figura B.4: *Opinion analysis tweet in italiano. Confronto fra i due sistemi di pesi.*

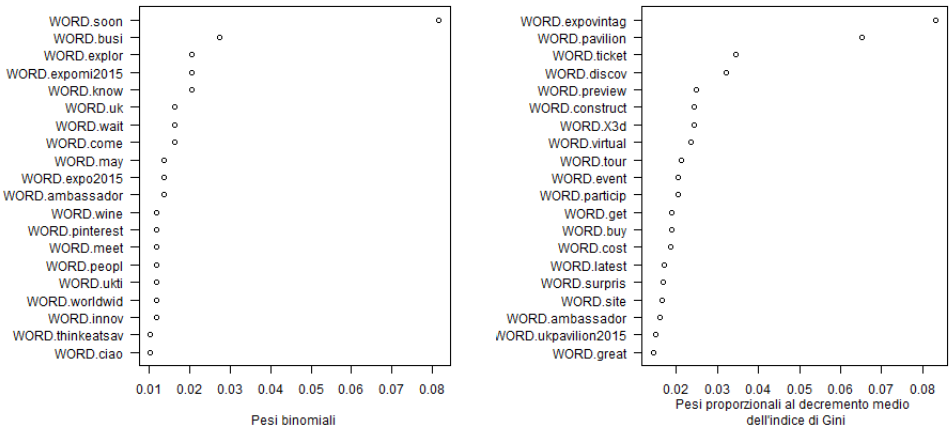


Figura B.5: *Opinion analysis tweet in inglese. Elenco dei 20 stem con pesi maggiori.*

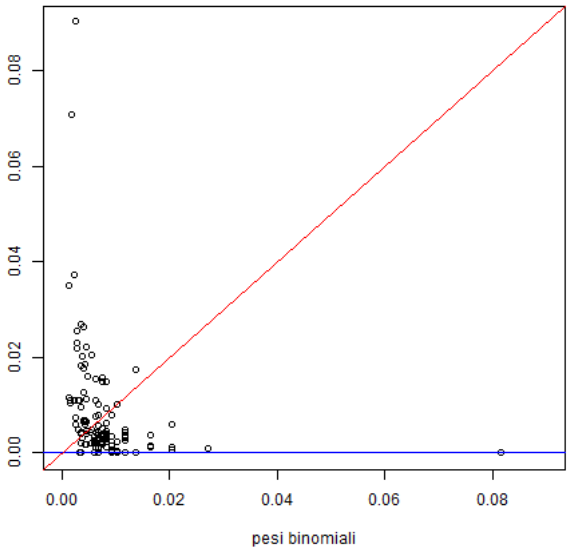


Figura B.6: *Opinion analysis tweet in inglese. Confronto fra i due sistemi di pesi.*

Quindi complessivamente si nota che i due sistemi di pesi sono completamente differenti, ma in base al modello risultano essere efficaci. Infatti nella sentiment analysis dei testi italiani si sono ottenuti dei risultati migliori con i pesi binomiali, mentre nell'opinion analysis con gli stessi testi oppure la sentiment analysis dei testi in inglese risultano migliori i classificatori con i pesi proporzionali al decremento medio dell'indice di Gini.

Certamente rimane una questione aperta e da indagare il perché un sistema di pesi come quello binomiale, in cui si da più importanza agli stem che compaiono o poche volte o tante volte, mentre si da meno peso agli stem che compaiono né troppo né poco, dia dei risultati così buoni.

Per indagare più affondo su quali siano gli stem ai quali viene assegnato un peso binomiale maggiore e anche perché si nota che in tutti i casi ci sono dei gruppetti che hanno uguale peso, è stata valutata la quantità $\frac{1}{q(1-q)}$ in cui q è la frequenza di uno stem in tutti i testi. Si osserva che di conseguenza l'espressione è proporzionale al peso che verrà assegnato allo stem. Si nota che le frequenze di tutte le parole in generale sono molto basse, quindi si assegnerà un peso alto agli stem che hanno una frequenza più bassa, basso a chi ha una frequenza più alta. Non ci sono infatti stem che abbiano frequenze altissime tali da poter avere dei pesi alti. Questo accade sia nei testi in italiano sia nei testi in inglese.

Appendice C

Stopwords italiane

Elenco delle stopwords italiane utilizzate in tutte le analisi fatte sui dataset in lingua italiana. È stato ricavato unendo vari elenchi di stopwords, quelli del pacchetto *tm*, quello del pacchetto *TextWiller* e altri elenchi vari trovati in rete.

a	avesti	c	dalla	eravate	facevo
abbia	avete	che	dalle	eri	fai
abbiamo	aveva	chi	dallo	ero	fanno
abbiano	avevamo	chissà	degl	essendo	farà
abbiate	avevano	ci	degli	etc	farai
ad	avevate	ció	dei	f	faranno
agl	avevi	ciò	del	fa	farebbe
agli	avevo	cmq	dell	faccia	farebbero
ai	avrà	coi	della	facciamo	farei
al	avrai	col	delle	facciano	faremmo
all	avranno	come	dello	facciate	faremo
alla	avrebbe	comunque	di	faccio	fareste
alle	avrebbero	con	dov	facemmo	faresti
allo	avrei	contro	dove	facendo	farete
anche	avremmo	cose	è	facesse	farò
anziche	avremo	così	e	facessero	fece
anziché	avreste	cui	é	facessi	fecero
anzichè	avresti	d	ebbe	facessimo	feci
avemmo	avrete	da	ebbero	faceste	fosse
avendo	avrò	dà	ebbi	facesti	fossero
avesse	avuta	dagl	ecc	faceva	fossi
avessero	avute	dagli	ed	facevamo	fossimo
avessi	avuti	dai	era	facevano	foste
avessimo	avuto	dal	erano	facevate	fosti
aveste	b	dall	eravamo	facevi	fu

fui
fummo
furono
g
già
gli
h
ha
hai
hanno
ho
i
il
in
io
j
k
l
la
le
lei
li
lo
loro
lui
m
ma
mi
mia
mie
miei
mio
n
ne
ne
né
nè
negl
negli
nei
nel
nell
nella
nelle
nello
noi
non
nostra
nostre
nostri
nostro
o
ogni
p
per
perche
perché
perchè
però
più
po
pò
poi
può
q
qual
quale
quali
quando
quanta
quante
quanti
quanto
quell
quella
quelle
quelli
quello
quest
questa
queste
questi
questo
r
s
sa
sarà
sarai
saranno
sarebbe
sarebbero
sarei
saremmo
saremo
sareste
saresti
sarete
sarò
se
sei
si
sia
siamo
siano
siate
siete
sono
st
sta
stai
stando
stanno
starà
starai
staranno
starebbe
starebbero
starei
staremmo
staremo
stareste
staresti
starete
starò
stava
stavamo
stavano
stavate
sarete
stavi
stavo
stemmo
stesse
stessero
stessi
stessimo
steste
stesti
stette
stettero
stetti
stia
stiamo
stiano
stiate
sto
su
sua
sue
sugl
sugli
sui
sul
sull
sulla
sulle
sullo
suo
suoi
t
ti
tra
tu
tua
tue
tuo
tuoi
tutti
tutto
u
un
una
uno
v
vabbè
vi
via
voi
vostra
vostre
vostri
vostro
w
xché
xchè
x
xche
xkè
y
z

Appendice D

Stopwords inglesi

Elenco delle stopwords inglesi utilizzato in tutte le analisi fatte sui dataset in lingua inglese.

i	them	does	you'll	who's	against	when
me	their	did	he'll	what's	between	where
my	theirs	doing	she'll	here's	into	why
myself	themselves	would	we'll	there's	through	how
we	what	should	they'll	when's	during	all
our	which	could	isn't	where's	before	any
ours	who	ought	aren't	why's	after	both
ourselves	whom	i'm	wasn't	how's	above	each
you	this	you're	weren't	a	below	few
your	that	he's	hasn't	an	to	more
yours	these	she's	haven't	the	from	most
yourself	those	it's	hadn't	and	up	other
yourselves	am	we're	doesn't	but	down	some
he	is	they're	don't	if	in	such
him	are	i've	didn't	or	out	no
his	was	you've	won't	because	on	nor
himself	were	we've	wouldn't	as	off	not
she	be	they've	shan't	until	over	only
her	been	i'd	shouldn't	while	under	own
hers	being	you'd	can't	of	again	same
herself	have	he'd	cannot	at	further	so
it	has	she'd	couldn't	by	then	than
its	had	we'd	mustn't	for	once	too
itself	having	they'd	let's	with	here	very
they	do	i'll	that's	about	there	

Elenco delle figure

Elenco delle tabelle

Bibliografia e Pacchetti R

Amati, G., Ambrosi, E., Bianchi, M., Gaibisso, C., and Gambosi, G. (2008). Automatic construction of an opinion-term vocabulary for ad hoc retrieval. *Advances in Information Retrieval, 30th European Conference on IR Research, Glasgow, UK. Proceedings.*

Amati, G., Bianchi, M., and Marcone, G. (2014). Sentiment estimation on twitter. *Proceedings of the 5th Italian Information RetrievalWorkshop, Pp.39-50.*

Blei, D. M., Ng, A. Y., and Jordan, M. J. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research 3, Pp. 993-1022.*

Bouchet-Valat, M. (09/08/2014). Pacchetto *SnowballC*: Snowball stemmers based on the C libstemmer UTF-8 library. *Versione 0.5.1.*

Branca, M. M. (2013/2014). *Strategie di Sentiment Analysis: confronti e nuove proposte.* Tesi di laurea, Università degli Studi di Padova.

Breiman, L. (2001). Random forests. *Machine Learning, 45, 5–32.*

Breiman, L. and Cutler, A. (17/07/2014). Pacchetto *randomForest*: Breiman and Cutler's random forests for classification and regression. *Versione 4.6-10.*

Breiman, L., Friedman, J., Stone, C. J., and Olshen, R. (1984). *Classification and Regression Trees.*

Cacco, F. (2012/2013). *Metodo Hopkins-King per la Sentiment Analysis: una valutazione basata sui tweets della campagna elettorale.* Tesi di laurea, Università degli Studi di Padova.

Ceron, A., Curini, L., and Iacus, S. M. (2014). Social media and sentiment analysis. l'evoluzione dei fenomeni sociali attraverso la rete.

Feinerer, I. (06/06/2014). Pacchetto *tm*: text mining package. *Versione 0.6.*

Feinerer, I. (25/01/2015). Pacchetto *wordnet*: WordNet interface. *Versione 0.1-10.*

Ferraccioli, F. (2013/2014). *Topic Model Workout: un approccio per l'analisi di microblogging, mass media e dintorni.* Tesi di laurea, Università degli Studi di Padova.

Gagolewski, M. and Tartanus, B. (11/12/2014). Pacchetto *stringi*: character string processing facilities. *Versione 0.4-1.*

Gentry, J. (11/02/2015). Pacchetto *TwitteR*: R based twitter client. *Versione 1.1.8.*

Girotto, A. D. (2012/2013). *Sentiment Analysis con il metodo Hopkings-King: studio dei commenti ad un post di un blog.* Tesi di laurea, Università degli Studi di Padova.

Go, A., Bhayani, R., and Huang, L. (2009). Twitter sentiment classification using distant supervision. *CS224N Project Report, Stanford, Pp.1-10.*

Hand, D. (2006). Classifier technology and the illusion of progress. *Statistical Science 21(1):1-14.*

Hopkins, D., King, G., Knowles, M., and Melendez, S. (20/08/2013). Pacchetto *ReadMe*: software for automated content analysis. *Versione 0.99836.*

Hopkins, D. J. and King, G. (2010). A method of automated nonparametric content analysis for social science. *American Journal of Political Science, Vol.54, No.1, Pp.229-247.*

King, G. and Lu, Y. (2008). Verbal autopsy methods with multiple causes of death. *Statistical Science 23(1):78-91.*

Laver, M., Benoit, K., and Garry, J. (2002). Extracting policy positions from political texts. using words as data. *The American Political Science Review, Vol. 97, No.2,Pp. 311-331.*

Melloncelli, D. (2012/2013). *Sentiment analysis in Twitter.* Tesi di laurea, Università di Bologna.

Pang, B. and Lee, L. (2005). Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *Proceedings of the ACL, Pp.79-86.*

Pang, B., Lee, L., and Vaithyanathan, S. (2002). Thumbs up? sentiment classification using machine learning techniques. *Proceedings of EMNLP, Pp.79-86.*

R Core Team (versione 3.1.1 del 10/07/2014b). R: A language and environment for statistical computing.

R Core Team (versione 3.1.2 del 31/10/2014a). R: A language and environment for statistical computing.

Rieder, B. (2013). Studying facebook via data extraction: the netvizz application. *In WebSci '13 Proceedings of the 5th Annual ACM Web Science Conference (pp. 346-355).*

Slapin, J. B. and Proksch, S. (2008). A scaling model for estimating time-series party positions from texts. *American Journal of Political Science, Vol. 52, No. 3, Pp. 705–722.*

Solari, D., Finos, L., Redaelli, M., Rinaldo, M., Branca, M., and Ferraccioli, F. (19/12/2013). Pacchetto *TextWiller*: collection of functions for text mining, specially devoted to the italian language. *Versione 1.0.*

Turney, P. D. (2002). Thumbs up or thumbs down? semantic orientation applied to unsupervised classification of reviews. *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL), Pp.417-424.*

van Rijsbergen, C., Robertson, S., and Porter, M. (1980). An algorithm for suffix stripping. *New models in probabilistic information retrieval. (British Library Research and Development Report, no. 5587).*

Wickham, H. (06/12/2012). Pacchetto *stringr*: make it easier to work with strings. *Versione 0.6.2.*

Wickham, H. (25/10/2014). Pacchetto *RSQLite*: SQLite interface for R. *Versione 1.0.0.*

Ringraziamenti

È arrivato il punto in cui dovrei stilare la lista delle persone da ringraziare, ma quando si fa una lista si rischia di dimenticare qualcuno oppure di mettere qualche nome di troppo. Così ho deciso di non metterne neanche uno.

Grazie a chi ha reso questi anni *milanesi* belli (sicuro che quelli che verranno saranno pure migliori!), grazie a chi mi ha fatto ridere e sorridere, ma soprattutto a chi mi ha sopportato, ha avuto pazienza e a chi mi ha aiutato nei momenti difficili.

Un grazie particolare va alla mia famiglia milanese (zii e cugini veri e quelli acquisiti), che mi ha accolto come un figlio e grazie ovviamente ai miei genitori e a mio fratello che nonostante tutto sono sempre al mio fianco, spero che questo sia solo uno dei tanti momenti di gioia che io vi possa regalare.

Mattia