Modelli Probabilistici per le Decisioni

-

NAÏVE BAYES ON TRIPADVISOR HOTEL REVIEWS

Mirko Rima 793435
Francesco Prete 793389
Andrea Spreafico 793317

Indice

Capitolo 1
Introduzione

Oggigiorno internet è una costante in ogni aspetto della nostra vita. Uno dei vantaggi principali che esso garantisce è la possibilità di avere a disposizione in ogni luogo e momento una quantità di dati, pareri e opinioni su praticamente qualunque prodotto o servizio. Questo ha portato a rendere una pratica comune, prima di acquistare qualcosa, controllare cosa altre persone pensano di quel prodotto o servizio. La possibilità di lasciare recensioni, infatti, è garantita in moltissimi siti proprio per l'importanza del parere degli acquirenti: esso è utile, in modi differenti, sia al venditore che all'acquirente. Le informazioni ricavate dalle recensioni possono essere usate per migliorare il proprio prodotto / servizio ed agire sul mercato nel migliore dei modi ma anche ad un nuovo possibile acquirente che vuole conoscere meglio il prodotto. Nella maggior parte dei siti, però, non è regolamentato il modo in cui lasciare una recensione e questa libertà può portare i recensori ad assegnare valutazioni complessive (che sono le uniche che vanno a concorrere nella valutazione finale dell'oggetto della recensione) non coerenti con i metadati e il testo della recensione. Questo può portare coloro che leggono in modo superficiale la recensione (o che non la leggono, affidandosi solamente alla valutazione complessiva) a conclusioni errate derivanti da informazioni condizionate dall'esperienza personale e soggettiva del recensore.

L'importanza di questo ambito e l'influenza che questo può avere da un punto di vista commerciale ci ha spinti ad effettuare un'analisi delle recensioni di hotel ottenute dal sito TripAdvisor.com

Capitolo 2
Scopo del progetto

Lo scopo principale di questo progetto è duplice: innanzitutto, dato il testo di una recensione e il valore dei metadati (i vari aspetti relativi all'hotel a cui è possibile assegnare una valutazione, come la pulizia, la location..)

- calcolare l'overall value più probabile per quelle evidenze
- successivamente, stimare la coerenza di questa recensione.

Allo scopo di rendere possibile un test del modello su nuove recensioni, è stato implementato un software che permette di inserire nuove recensioni (con testo e metadati) pronte per essere analizzate. Nel paragrafo precedente si è parlato di "coerenza" di una recensione. Daremo ora una definizione di quello che, in questo specifico progetto, si intende quando si utilizza questo termine:
"una recensione è coerente se non esiste una differenza significativa tra la valutazione complessiva assegnata dal recensore e quella che, sulla base di testo e metadati, darebbe un valutatore esterno".
L'idea alla base di questa definizione è che un recensore, nella fase di decisione della valutazione complessiva di una struttura sia condizionato dall'esperienza personale vissuta nell'hotel e che quest'ultima possa modificare (anche pesantemente) il voto, nonostante le singole categorie siano effettivamente risultate buone valutate singolarmente.

Se è vero che per molti è comprensibile e utile un approccio del genere, è anche vero che un lettore poco attento che sta per affrontare un'esperienza completamente diversa da quella del recensore (magari un viaggio di lavoro mentre nella recensione si parlava di una vacanza) potrebbe volere un semplice parere oggettivo sulla struttura senza l'influenza di situazioni che non vivrà nello stesso modo del recensore.

Capitolo 3
Scelte implementative

Il software è stato implementato in Python (v 3.6.3) utilizzando alcune librerie a supporto dello sviluppo. In particolare, per ognuna delle 5 fasi, abbiamo utilizzato una diversa libreria:

- pandas - per l'analisi dei dati e la creazione dei dataset
- nltk - per il preprocessing del contenuto testuale delle recensioni
- pgmpy - per la generazione della rete bayesiana e l'inferenza *
- scikit-learn - per la valutazione delle performance del modello
- tkinter - per la gestione dell'interfaccia grafica

Grazie a questo software è possibile testare entrambi gli obiettivi del progetto: inserendo come input una nuova recensione, infatti, verrà stimato l'overall value (eseguendo il task di inferenza sulla rete) e verrà controllato che la recensione sia coerente.

* **N.B.** Nonostante l'utilizzo di pgmpy, il calcolo delle CPT è stato eseguito tramite script e non con la funzione standard della libreria. Questo è stato necessario poiché la funzione interna alla libreria non computava le CPT correttamente secondo il criterio della maximum likelihood estimation.

Capitolo 4
Descrizione del Dataset

Dataset Iniziale:

- Reviews.csv - contiene circa 135 mila recensioni di hotel. Dove per ogni recensione sono riportati alcuni dati generali (id, autore..) ed una serie di metadati rappresentati nella tabella sottostante

Attributo	Tipo	Descrizione	Range
Review_ID	Int	Identificativo della recensione	/
Hotel_ID	Int	Identificativo dell'Hotel	/
Author	String	Nome dell'account dell'autore della recensione	/
Content	String	Testo della recensione	/
Date	Date	Data della recensione	2003 - 2008
No. Reader	Int	Numero di lettori della recensione	/
No. Helpful	Int	Numero di giudizi utili della recensione	/
Overall	Int	Valutazione Complessiva riguardante l'esperienza dell'utente	1 - 5*
Value	Int	Valutazione del valore (Rapporto Qualità / Prezzo)	1 - 5
Rooms	Int	Valutazione delle stanze	1 - 5
Location	Int	Valutazione della posizione	1 - 5
Cleanliness	Int	Valutazione della pulizia	1 - 5
Check in / front desk	Int	Valutazione del check in	1 - 5
Service	Int	Valutazione del servizio	1 - 5
Business service	Int	Valutazione del servizio di business	1 - 5

* Nel dataset in tutti i valori Int se presente il valore -1 è considerato come NULL (Missing Information)

Durante la prima fase abbiamo analizzato il dataset Reviews.csv riscontrando tre problemi principali:

- Esistenza di colonne non rilevanti per lo scopo di questo progetto, che sono state eliminate
- Alta presenza di valori nulli nei metadati, risolto mantenendo solo le istanze con al più 3 valori nulli (poiché dopo l'analisi del dataset, abbiamo scoperto che troppi metadati nulli potevano trarre in inganno il training del modello)
- Uno squilibrio nella distribuzione delle classi risolto con una tecnica di under-sampling per bilanciare il dataset

Dataset Finale:
Successivamente sono state selezionate 6500 istanze random per ogni classe e il dataset è stato diviso in Training e Test (70% e 30% rispettivamente), mantenendoli entrambi bilanciati. Il nuovo dataset è mostrato nella tabella sottostante.

Attributo	Tipo	Descrizione	Range
Overall	Int	Valutazione Complessiva riguardante l'esperienza dell'utente	1 - 5
Value	Int	Valutazione del valore (Rapporto Qualità / Prezzo)	1 - 5
Rooms	Int	Valutazione delle stanze	1 - 5
Location	Int	Valutazione della posizione	1 - 5
Cleanliness	Int	Valutazione della pulizia	1 - 5
Check in / front desk	Int	Valutazione del check in	1 - 5
Service	Int	Valutazione del servizio	1 - 5
Business service	Int	Valutazione del servizio di business	1 - 5
T:$term_1$	Boolean	Booleano che indica la presenza o l'assenza del termine ennesimo nella recensione*	0/1
...	...	...	...
T:$term_N$	Boolean	Booleano che indica la presenza o l'assenza del termine ennesimo nella recensione	0/1

* Per ogni Term selezionato sarà presente una colonna nel nostro dataset

Esplorazione dei dati

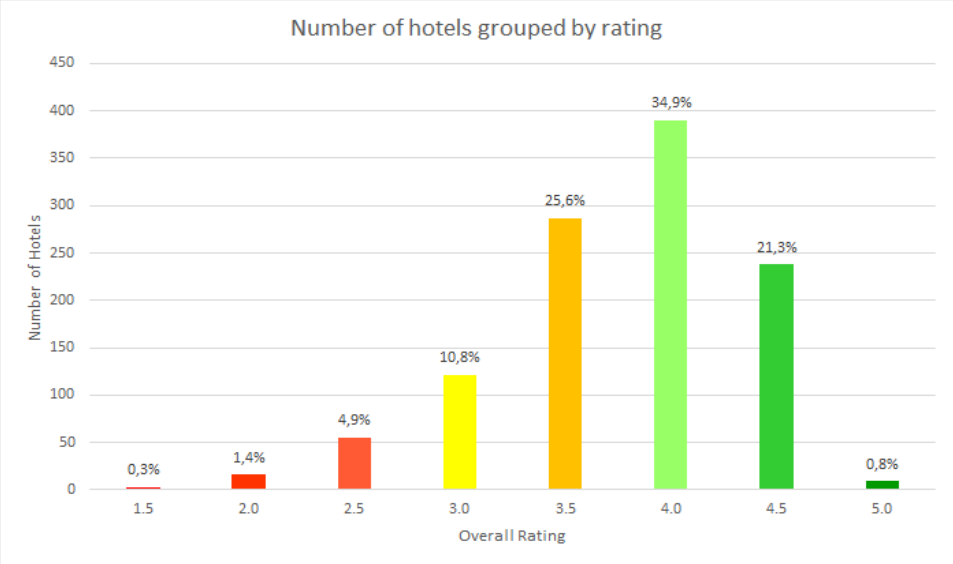
Prima di procedere con le successive fasi di analisi e sviluppo è stato analizzato il dataset generato. Un'analisi esplorativa è stata necessaria in modo da comprendere meglio i dati. Questa fase, divisa secondo diversi parametri è riporta di seguito. Tutte le statistiche sono state effettuate utilizzando python (v 3.6.3).

Number of reviewed hotels: 1119

Number of reviews: 32500

Number of hotels grouped by rating:

Number of hotels grouped by rating	
Overall Rating	**Number of Hotels**
1.5	3
2.0	16
2.5	55
3.0	121
3.5	287
4.0	390
4.5	238
5.0	9

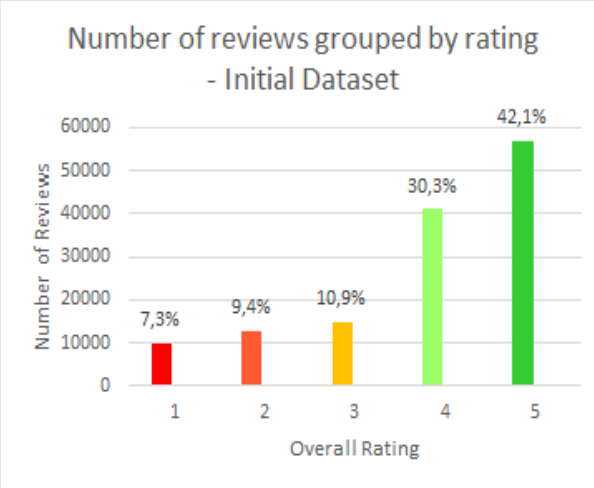


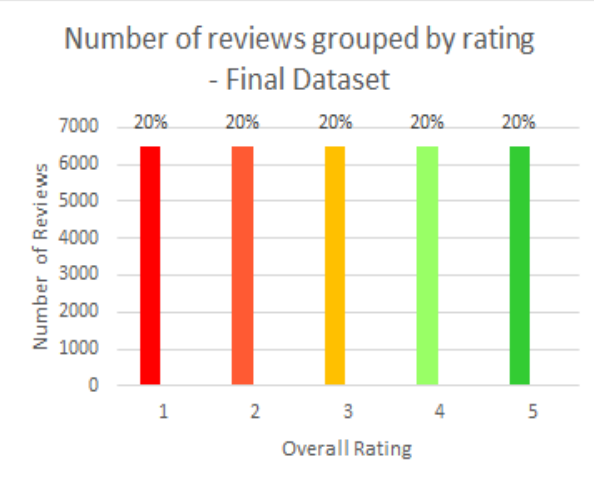
Group of hotels by average price interval:

Group of hotels by average price interval						
Range	0 - 100	100 - 200	200 - 300	300 - 400	400 - 500	>500
Number of hotels	73	434	349	141	48	52

Number of reviews grouped by rating:

	Number of Reviews	%	Number of Reviews	%
Overall Rating	Initial Dataset		Final Dataset	
1	9869	7,3%	6500	20%
2	12729	9,4%	6500	20%
3	14744	10,9%	6500	20%
4	41063	30,3%	6500	20%
5	56913	42,1%	6500	20%





Capitolo 5
Analisi del testo

Una volta ottenuti i dataset finali per il training del modello, è stata effettuata una fase di analisi del contenuto testuale delle recensioni al fine di ottenere una rappresentazione vettoriale di quest'ultimo. Durante la parte iniziale dell'analisi abbiamo utilizzato una combinazione di metodi della classe string di Python e NTKL (Natural Language ToolKit) con lo scopo di:

- Mettere il testo in minuscolo
- Rimuovere i caratteri non codificati in ASCII, la punteggiatura, i numeri e le stopword
- Tokenizzare il testo (ovvero la divisione in token tali che ogni token fosse una diversa parola contenuta nella recensione)
- Stemmare il testo (ovvero il processo di riduzione della forma flessa di una parola alla sua forma radice)

A questo punto era necessario ridurre il numero di parole e si è deciso di considerare solo le 500 parole più frequenti. Per ognuna di queste è stata calcolata (grazie a senticnet) la polarità: le parole con polarità superiore (in modulo) ad una certa soglia sarebbero state mantenute, altrimenti scartate. Il risultato sono le 74 parole sotto riportate.

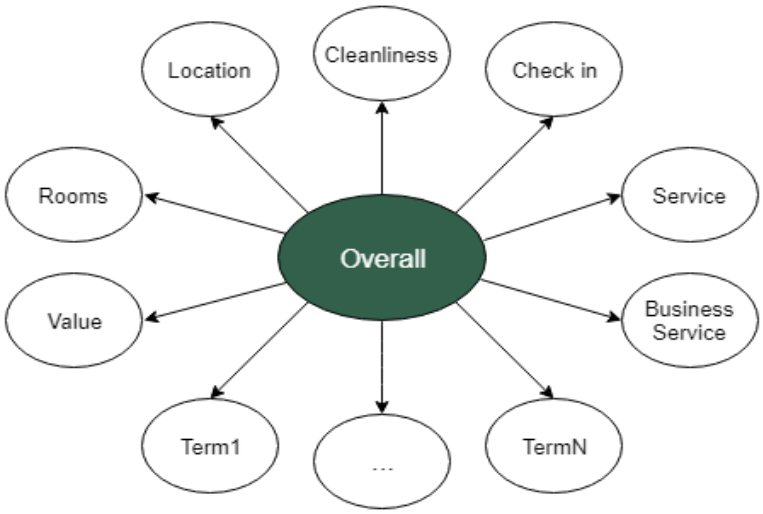
pool, shuttle, air, block, kind, music, within, entire, taxi, next, horrible, better,ended, right, please, experience, card, sure, person, terrible, concierge,happy, friendly, modern, problem, including, pretty, morning, cleaned, dirty, disappointed, fantastic, enjoy, enough, perfect, money, customer, resort, expensive, stop, quiet, fun, great, extremely, view, spa, space, without, good, quality, entertainment, party, least, service, star, rude, course, show, love, back, help, far, special, hard, tried, definitely, easy, wonderful, give, pay, liked, bad, worst

In ultimo, per ognuna delle parole è stata aggiunta una colonna (sia nel training set che nel test set) con un valore booleano uguale a 1 per indicare la presenza della parola nella recensione o uguale a 0 per indicare che la parola non è presente nella recensione.

Capitolo 6
Descrizione del modello

A questo punto si è passati alla fase di implementazione del modello di classificazione. La scelta è ricaduta su una rete di tipo Naive Bayes. Questo modello consistente in un particolare tipo di Bayesian Network, in cui la rete prevede un solo nodo padre, considerato la causa comune che spiega il verificarsi di tutti gli altri nodi figli. Inoltre il modello richiede la conoscenza di tutte le probabilità a priori e condizionali relative al problema e, per questo motivo, nel capitolo precedente abbiamo illustrato il procedimento necessario al calcolo delle CPT. Considerando la struttura della rete e delle CPT, nel modello teorico tutti i nodi figli devono rispettare la relazione di indipendenza condizionale tra loro. Nel nostro caso, è intuivo come tutti i metadati siano effettivamente condizionalmente indipendenti tra loro, infatti una buona location per una struttura è indipendente dal grado di pulizia delle camere o dalla cortesia del personale.

Il nodo padre all'interno della rete è la variabile target (overall value) e i figli sono i 7 metadati più una serie di valori booleani che indicano la presenza (o l'assenza) di alcune parole specifiche derivanti dall'analisi testuale delle recensioni riportata nel capitolo successivo.

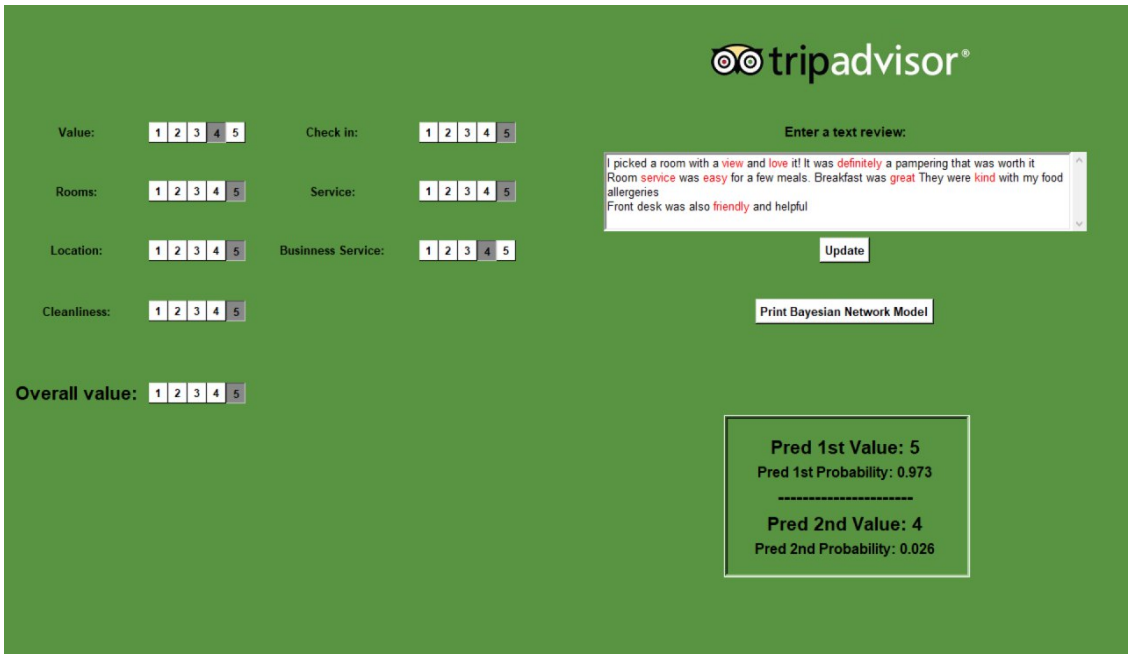


Capitolo 7
Software implementato

Il software implementa un form di creazione di una recensione, basato sulla struttura utilizzata da TripAdvisor, fornendo la possibilità di assegnare un valore ai parametri necessari a recensire un hotel. Questi parametri si suddividono in due categorie:

1. **Metadati:** l'utente può scegliere un valore tra 1 e 5 oppure nullo (non cliccando nessuna delle precedenti opzioni), a patto che non ci siano più di 3 valori nulli al momento della creazione della recensione
    - Value: rapporto qualità/prezzo
    - Rooms: qualità della camera assegnata al recensore
    - Location: qualità della posizione dove è situato
    - Cleanliness: grado di pulizia della camera e della struttura
    - Check In: qualità del servizio di check-in
    - Service: qualità dei servizi offerti
    - Business Service: qualità dei servizi per business
    - Overall Value: valutazione complessiva, ovvero considerando tutti i fattori influenti, riguardante la permanenza del recensore presso l'hotel.
2. **Recensione testuale:** l'utente può inserire un testo
    - Text Content: recensione testuale che permette all'utente di esprimere osservazioni, giudizi o esperienze rilevanti non riportabili tramite i parametri considerati dai metadati.

Una volta inseriti i dati, il software permette di eseguire un task di inferenza sulla rete bayesiana precedentemente implementata. Il risultato di questo task è la predizione dei due valori più probabili dell'overall value e le relative probabilità considerando i dati inseriti dall'utente. Di seguito è riportata un'immagine raffigurante l'interfaccia grafica del software sopra descritto.



Capitolo 8
Analisi dei Risultati

A questo punto siamo passati alla fase di testing delle performance del modello. Per fare ciò abbiamo eseguito il task di inferenza sulle istanze del test set confrontando l'overall value reale con quello predetto.

Accuracy	Recall	Precision	F1 score
0.668	0.668	0.674	0.666

E' possibile notare come l'accuracy del modello non sia particolarmente alta. Notiamo inoltre che precision e recall siano molto simili al valore dell'accuratezza indicando che il modello non ha particolari bias di decisione. Di conseguenza anche l'F1_Score non si discosta significativamente da questi valori.

Le performance risultanti sono causate principalmente della natura del problema: l'overall value inserito da un recensore non è un dato oggettivo ma una percezione generale dell'esperienza vissuta in un hotel. Come descritto in precedenza, una recensione molto soggettiva è legittima, ma potrebbe indurre in un'errata valutazione generale della struttura.

Recensioni di questo tipo, inducono in errore non solo un possibile lettore ma anche il classificatore perchè spesso non esistono dati per poter comprendere le motivazioni di tale valutazione. Per ovviare almeno parzialmente a questo problema, abbiamo deciso di provare un approccio multi-label, utilizzando non solo la classe considerata più probabile dal modello, ma anche la seconda scelta (2-Labels classification). Questo approccio ci ha permesso di evitare una scelta arbitraria in tutti i casi dubbi rendendo molto più precisa la predizione. La giustificazione per questa scelta è data dal fatto che, spesso, anche un valutatore umano sarebbe in dubbio sul voto da assegnare ad una recensione: casi in cui la varianza tra i metadati è molto alta, o il testo si discosta di parecchio dal valore medio dei metadati sono difficilmente valutabili da chiunque.

L'accuratezza, dopo queste assunzioni, è infatti aumentata dal 66,6% al 91,5%. E' necessario fare una precisazione: andando a valutare la media dei metadati e l'overall della recensione, si è notato come il rimanente 8,5% delle recensioni non è composto solo da errori di classificazione ma anche da errori utente. In generale abbiamo rilevato $\frac{1}{3}$ degli errori è legato alla rete, mentre $\frac{2}{3}$ ad errori utente. Abbiamo considerato errori della rete recensioni in cui il grado di coerenza tra la valutazione

del recensore e la recensione (metadati, testo e overall value) era elevato ma la rete, per qualche motivo, non ha individuato il corretto valore. Invece, quando overall value e recensione non erano coerenti, abbiamo considerato l'errata classificazione un errore imputabile all'utente. All'interno di questi errori abbiamo individuato due pattern nei dati: errori di incoerenza ma anche missclick. Queste recensioni presentano discrepanze così alte tra valor medio dei metadati e overall value e un grado di incoerenza tra il testo e la valutazione finale talmente alto da non essere giustificabile nemmeno dall'esperienza soggettiva vissuta dal recensore nell'hotel, ma imputabile quasi certamente ad un missclick. Trascurando le istanze, il cui errore è imputabile all'utente, l'accuratezza stimata per la rete a 2-labels è del 97%, aumentando del 6,5%.

Nella tabella sottostante sono riportati degli esempi esplicativi delle 3 categorie di errore sopra descritto.

	Errore utente		**Errore della rete**
	Missclick	**Incoerenza**	
Value	5	5	2
Rooms	4	5	2
Location	4	5	5
Cleanliness	4	5	2
Check -in	4	4	5
Service	4	5	2
Business service	-1	5	3
Review Content	• "It is extremely romantic and relaxing." • "All in all, the Excellence Punta Cana is a heaven." • "I have never written a review but my stay was so enjoyable I just wanted to share."	• "Fantastic hotel!" • "Everything about this hotel is good -food/room /pool/bar/parking etc." • "This really is 5 star with value for money. Oh except for the phone calls"	• "We asked for a room upgrade, and got one for free. The customer service at this resort is amazing!" • "There were these little bugs all over the room [...] but it was disgusting!" • "Outside of the room, the resort was BEAUTIFUL!" • "So, in conclusion...it was hard to rate this experience overall... for the people and the resort grounds were beautiful...the rooms and travel agency was horrible, and the food was okay."
Overall value	**1**	**3**	**3**
Predicted value	**4**	**5**	**2**

Capitolo 9
Conclusioni

Ricordiamo brevemente lo scopo di questo progetto: implementare un sistema in grado di, data una recensione con testo e metadati, classificare correttamente la valutazione complessiva di quest'ultima e indicare se la recensione potrebbe essere incoerente.

Per quanto riguarda la prima fase, Naive Bayes ha ottenuto risultati discreti e l'accuratezza è notevolmente aumentata nella classificazione multi-label grazie anche alla doppia possibilità di avere una risposta corretta, giustificata però da un tipo di problema per cui è spesso difficile avere una singola risposta oggettiva.

La fase di rilevazione delle incoerenze, ha anch'essa ottenuto buone performance e potrebbe essere un buon sistema per supportare il controllo delle recensioni da parte di un sito che permette ad utenti di lasciare recensioni relative ad hotel, segnalando possibili recensioni non coerenti allo scopo di utilizzare solo quelle più coerenti e oggettive possibile per la determinazione del voto finale di una struttura.