Amazon Reviews

Sentiment and Aspect Based Analysis

Coppola Matteo	Palazzi Luca	Vivace Antonio
793329	793556	793509

Data Analytics, January 2020

Abstract

Recentemente il mercato dello shopping online sta acquisendo sempre più rilevanza, superando i limiti della compravendita in negozi fisici e in alcuni ambiti rimpiazzandola. Con la crescita degli acquisti, cresce anche la mole di dati che venditori, produttori, pubblicitari e gestori di piattaforme di e-Commerce si trovano a dover processare per ottenere informazioni sulla natura delle transazioni, dei clienti che le producono e sui trend mercato.
Una parte fondamentale di questi dati è costituita da quelli prodotti dai consumatori stessi dopo aver effettuato l'acquisto: opinioni, recensioni e valutazioni sul prodotto ed in generale sull'esperienza di acquisto.
Sentiment Analysis è una materia che sfrutta dati di questa natura (denominati VOC: *Voice of the Costumer*) per estrarre, quantificare e studiare informazioni soggettive in modo sistematico.
Tra i campi che beneficiano di questi strumenti troviamo: sviluppo di strategie di marketing, sistemi di raccomandazioni, *brand monitoring*, servizio clienti e ricerche di mercato.
In questo lavoro, analizziamo un insieme di recensioni pubblicate su Amazon su articoli della categoria "Cellulari e accessori correlati" per uno studio esplorativo, estraendo dati statistici ed evoluzioni temporali sulla natura delle recensioni e delle valutazioni numeriche annesse. Procediamo poi nell'addestrare modelli di Machine Learning (Logistic Regression e Naive Bayes) per valutare la loro efficacia nell'identificare correttamente il sentimento generale delle recensioni, analizzandone poi le metriche e cercando di individuare i migliori iperparametri. Infine per i sei prodotti più recensiti applichiamo una tecnica di Topic Analysis per individuare i cluster di argomenti.

Contents

List of Figures

List of Tables

Chapter 1

Introduzione

Negli ultimi decenni, l'avvento e la popolarizzazione di servizi online ha cambiato il volto dello shopping su larga scala. Piattaforme come Amazon ed eBay fanno parte della vita di tutti i giorni ed è frequente consultare risorse online prima di acquistare. Nel 2017, il volume di vendite nel mercato statunitense che vengono effettuate online ha raggiunto il 9% e ci si aspetta che arrivi al 12% nel 2021 [1]. La crescita del traffico e della portata dei portali di commercio online genera una quantità crescente di dati sulla natura delle transizioni e degli utenti di questo servizio.
Una parte importante di questi dati è costituita dai contenuti generati dagli utenti che valutano i prodotti acquistati e condividono la loro esperienza. Si tratta principalmente di valutazioni numeriche, spesso corredate da un breve paragrafo testuale.
Avere a disposizione un insieme di strumenti che possa processare in modo automatico questa mole di dati è fondamentale per tutti gli attori coinvolti nelle transazioni: produttori, clienti/consumatori, venditori, piattaforme di vendita e pubblicitari.
Discipline come la Sentiment Analysis estraggono dei dati strutturati da questi contenuti testuali, permettendo uno sguardo statistico sulle tendenze di comunità di acquirenti sotto diversi aspetti di diversi prodotti. Avere un'idea di quali siano gli elementi più o meno apprezzati di un prodotto, secondo le diverse categorie di utenti permette di agire in modo dinamico e veloce sul loro sviluppo e sulla loro pubblicizzazione. I gestori di questi portali invece saranno interessanti a profilare gruppi di utenti, estraendone le preferenze, e gli elementi di successo dei prodotti, per proporre raccomandazioni sempre più vincenti, accurate e vicine ai desideri dell'utente.
Un altro aspetto da non sottovalutare è quello del valore "genuino" che i contenuti generati da altri consumatori riescono a trasmettere. Le recensioni vengono infatti recepite come fonti affidabili e privi di natura pubblicitaria, rappresentando uno strumento molto potente.
Amazon ha sviluppato un sistema per assegnare rilevanza alle recensioni e non è raro che venga usato insieme ad altre tecniche di (auto) marketing, promuovendo

articoli con recensioni positive e utili, presentandole ordinate dalla più "convincente" all'utente che sta attraversando il processo di decisione.

1.1 Obiettivo del progetto

Questo lavoro si sviluppa in tre fasi di seguito sintetizzate.

1.1.1 Esplorazione

Per approfondire e comprendere la natura di questi contributi, sono state effettuate analisi preliminari sulle recensioni, concentrandoci sulla distribuzione delle opinioni, sulle caratteristiche delle recensioni sui prodotti più rilevanti e sull'evoluzione di questi ultimi fattori nel tempo.
L'obbiettivo è sviluppare una visione su diversi aspetti soggettivi (e variabili nel tempo e per categoria) che caratterizzano le recensioni, al fine di comprendere in che modo vengano prodotte ed interpretate.
Abbiamo approfondito caratteristiche come `verified` e in che modo influenzano il totale dei dati.
Infine, vengono brevemente presentate alcune ricerche che investigano la questione dello sbilanciamento delle recensioni.

1.1.2 Sentiment Analysis

In questa fase, analizziamo sistematicamente le parti testuali delle recensioni per estrarne un'opinione.
Una parte preliminare pre-processa e prepara il dataset. Vengono scartate recensioni prolisse e ritenute inutili e fatte ulteriori esplorazioni sul nuovo (ristretto) corpo di recensioni.
Infine, alleniamo due classificatori, Naive Bayes e Logistic Regression, che etichettano queste istanze con la variabile target `opinion` e valutiamo le loro performance.

1.1.3 Topic Analysis

In questa fase, viene utilizzato un algoritmo che consente di identificare gli argomenti più discussi all'interno di un corpus di documenti.
La fase di preparazione del dataset è la stessa della fase di sentiment analysis.
Vengono inoltre analizzati gli svantaggi e alcune possibili soluzioni del metodo analizzato. Infine, gli argomenti risultanti dalla sua applicazione vengono visualizzati in maniera interattiva.

1.2 Dataset

Il dataset utilizzato [8] proviene da un gruppo di ricerca dell'Università di San Diego, che ha estratto e processato le recensioni rilasciate dagli utenti sul sito

Amazon.com fino al 2018 in formato JSON.
Abbiamo scelto il dataset della categoria "Cellulari ed Accessori", in una versione densa, contenente solo i dati generati da utenti con almeno 5 recensioni (*5-core*).

1.3 Software utilizzati

Python è stato lo strumento fondamentale in questo lavoro, scelta dovuta alla grande quantità di strumenti e librerie open source disponibili per questo linguaggio.

Tra le librerie utilizzate, ricordiamo:

- Pandas per il caricamento, manipolazione e querying dei dataset
- Matplotlib per il rendering di grafici e figure direttamente da dataframe pandas
- numpy per un supporto efficiente a matrici e vettori di grosse dimensioni
- sklearn per machine learning
- pyLDAvis per la visualizzazione interattiva dei topic model
- NLTK per Natural Language Processing
- VueJS per applicazioni web reattive
- Flask per realizzare un'API Restful con le funzionalità implementate

Il versionamento del codice e la possibilità di lavorare in gruppo sono due importanti strumenti offerti da Git, mentre la documentazione è scritta in LaTeX.
I prodotti del progetto sono: script per ogni fase della pipeline, Notebook Jupyter interattivi, figure e grafici vettoriali ed un'applicazione web composta da un backend in Python e un frontend in Vue.js che offre un un'interfaccia utente di facile utilizzo che espone alcune funzionalità del nostro lavoro.
Inoltre, per lo sviluppo della demo sono state utilizzate tecnologie frontend basate su Javascript.

Chapter 2

Esplorazione dei dati

2.1 Informazioni preliminari sul dominio

Prima di addentrarci nell'analisi del nostro dataset, che si limita ad una categoria, abbiamo cercato delle visualizzazioni globali dell'intero marketplace Amazon. La Figura 2.1 ci mostra una carattere fortemente stagionale: gli utenti sono molto più propensi a fornire recensioni nei periodi estivi, nonostante i picchi dei volumi di vendita si verifichino intorno al periodo natalizio [10].

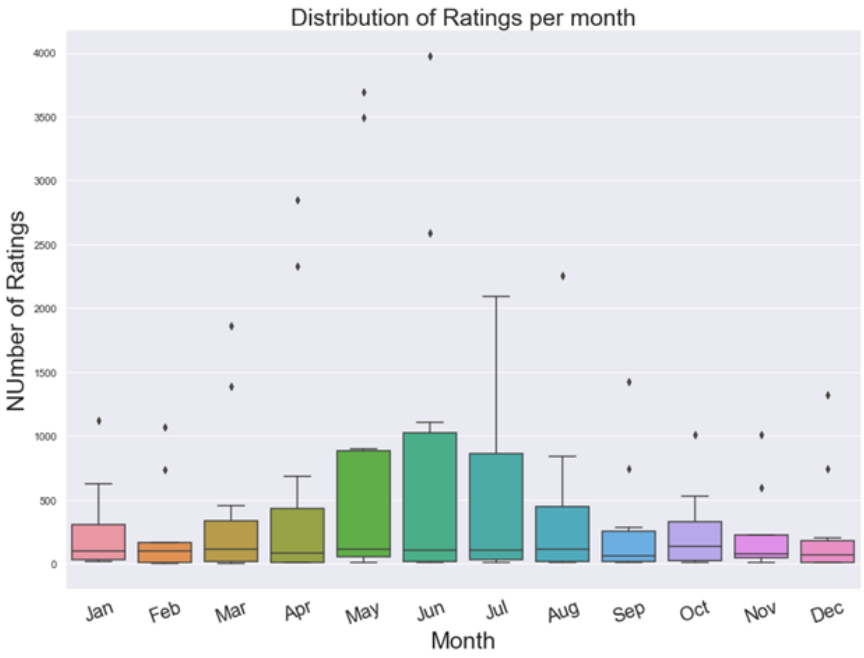


Figure 2.1: General Amazon ratings per month [9]

La Figura 2.2 visualizza invece il contributo di un utente, dandoci un'idea di quanto vocale sia la clientela Amazon, in media.

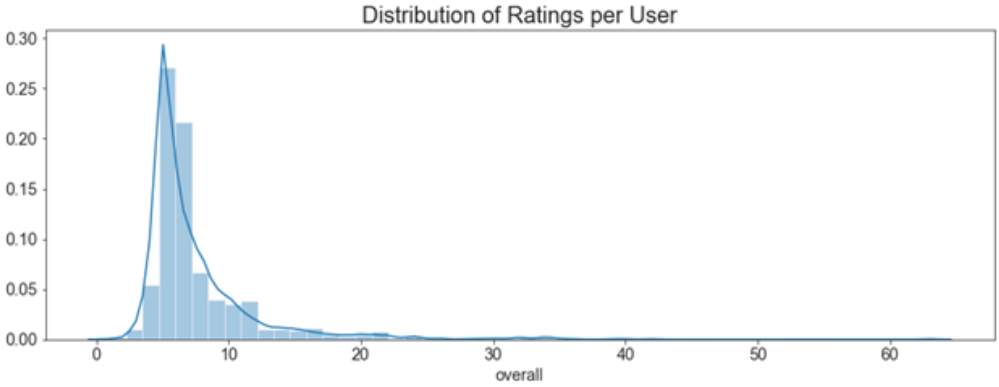


Figure 2.2: General Amazon ratings per user [9]

2.2 Descrizione dataset

Il dataset si presenta in formato JSON e viene caricato in memoria in un DataFrame con la libreria Pandas, molto efficiente per la gestione di dati voluminosi.
La fase di caricamento e preprocessamento del dataset sono le più impegnative computazionalmente, impiegando gran parte del tempo totale.
Per ovviare a questo problema e muoverci più agevolmente durante lo sviluppo sfruttiamo la funzione `to_pickle` di Pandas per salvare su disco una versione "cachata" del dataframe, abbreviando le successive esecuzioni della pipeline.
Prima del salvataggio sono state effettuate alcune operazioni utili per rendere il dataset conforme agli obiettivi. In particolare:

- Il campo `vote` è stato trasformato da tipo `object` a tipo `float`
- Il campo `reviewText` possedeva alcune recensioni vuote, inutili e perciò eliminate

Queste operazioni hanno ridotto il dataset portandolo da un totale di recensioni pari a 1128437 a un totale di 1127654, suddivise fra ben 157195 utenti e 48146 prodotti.

Campo	Tipo	Descrizione
overall	int	Valutazione del prodotto (1-5)
verified	bool	Recensione proveniente da acquisto verificato
reviewTime	string	Data della recensione in formato string
reviewerID	string	Codice univoco del recensore
asin	string	Codice univoco del prodotto
style	string	Dizionario dei metadati del prodotto
reviewerName	string	Nome del recensore
reviewText	string	Testo della recensione
summary	string	Titolo della recensione
unixReviewTime	int	Data della recensione in formato unix
vote	float	Numero di voti della recensione
image	string	Immagine associata alla recensione

Table 2.1: Campi del dataset con tipo e descrizione

Il dataset possiede gli attributi mostrati in Tabella 2.1. Ogni record del dataset è la rappresentazione di una singola recensione svolta da parte di un utente per un certo prodotto nella data indicata.
Per l'identificazione dell'utente abbiamo a disposizione il campo `reviewerName` e il campo `reviewerID`: utilizzeremo solamente quest'ultimo per i nostri scopi.
Per quanto riguarda i campi relativi alla recensione, abbiamo a disposizione sia `summary` che `reviewText`.
Per identificare il prodotto abbiamo a disposizione solamente il campo `asin`, che è un codice univoco da cui si può risalire a maggiori informazioni con l'utilizzo delle API Amazon o software di terze parti.
Le recensioni sono classificate come *verificate* se provengono da un acquisto su Amazon per almeno l'80% del valore originale dell'articolo. L'utente deve aver inoltre speso almeno 50$ sul proprio account.

2.3 Estensione del dataset

A partire dal dataset originale abbiamo creato dei nuovi campi ritenuti di valore per effettuare una fase di esplorazione più approfondita.

2.3.1 Da `overall` a `opinion`

Osservando la distribuzione del campo `overall`, mostrata in Figura 2.3, possiamo notare un forte sbilanciamento sul valore 5: questa tendenza è presente anche in dataset Amazon di categorie diverse dalla nostra.

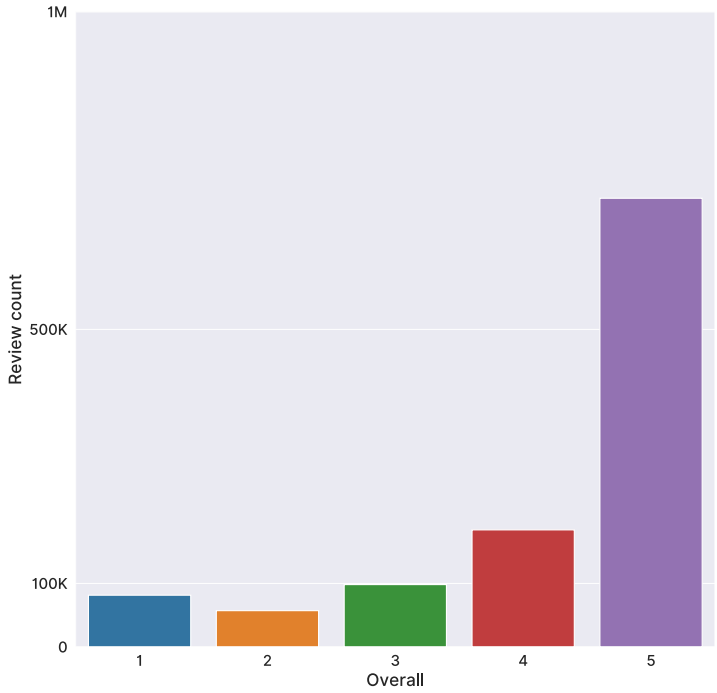


Figure 2.3: Overall distribution

In previsione della fase di sentiment analysis, il campo `overall` è stato utilizzato per la creazione del campo `opinion`, così composto:

- I valori 1 e 2 vengono trasformati in *negative*
- Il valore 3 viene trasformato in *neutral*
- I valori 4 e 5 vengono trasformati in *positive*

In Figura 2.4 viene mostrata la distribuzione: essa è ovviamente simile a quella già osservata per il campo `overall` e sarà quindi necessario un bilanciamento del dataset per la fase di sentiment analysis.

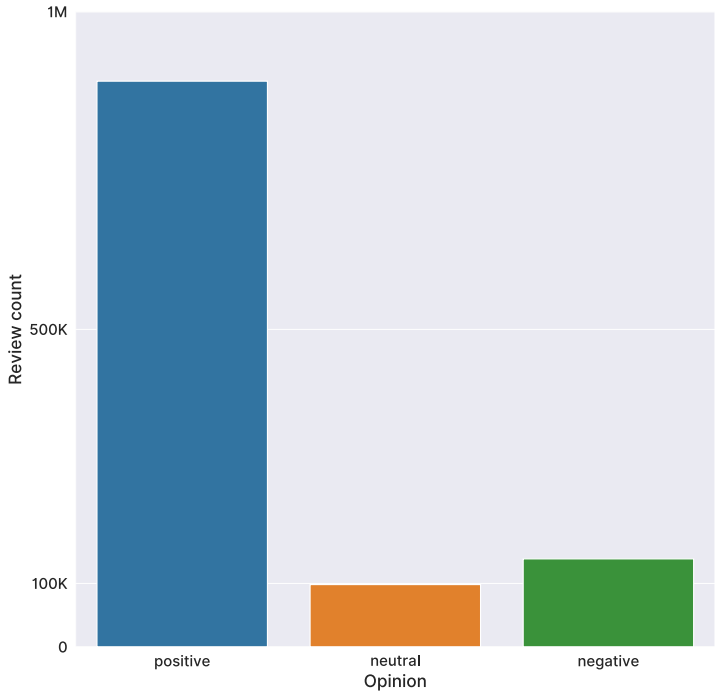


Figure 2.4: Opinion distribution

2.3.2 Conteggio delle parole nelle recensioni

Il campo `reviewText` è di fondamentale importanza per le fasi di sentiment e topic analysis. Ma per la fase di esplorazione, essendo il testo di una recensione un dato qualitativo, non è di alcun valore. Per questo motivo, abbiamo computato direttamente il numero di parole e creato il campo risultante `n_words`. In Figura 3.5 viene mostrata la distribuzione del campo `n_words` rispetto al campo `opinion`, tenendo in considerazione solamente le recensioni con meno di 1000 parole per una questione di visibilità che sarebbe venuta meno considerando anche le (poche) recensioni composte da oltre 1000 parole.

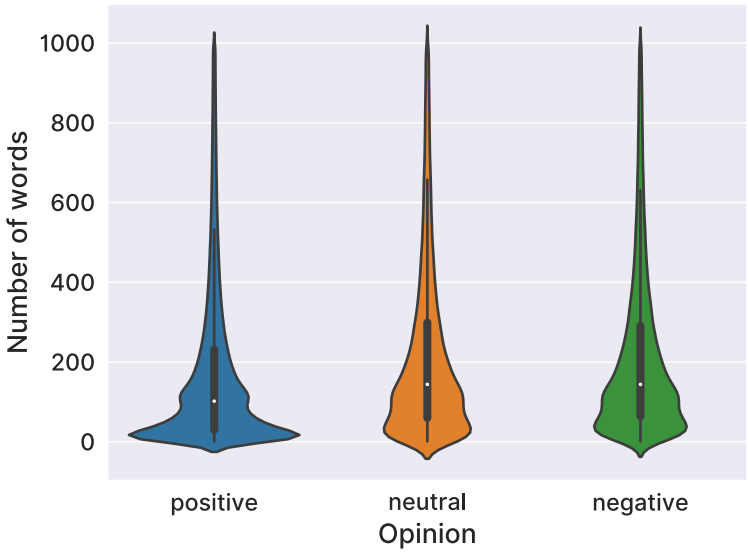


Figure 2.5: Distribution of words in review for each opinion

2.3.3 Analisi temporale

Il campo `unixReviewTime` fornisce la data della recensione in formato unix. Con alcune semplici manipolazioni del suddetto campo abbiamo creato i seguenti:

- `month_year` nel formato YYYY-MM
- `month` nel formato MM
- `year` nel formato YYYY
- `week_day` in cui il giorno della settimana è rappresentato con un numero intero (0-6)

Il dataset considera recensioni nell'arco di 16 anni circa: più precisamente la prima recensione risale al 23-10-2002, mentre l'ultima al 01-10-2018. Considerato il dominio trattato, un'analisi di valore è quella di considerare la distribuzione delle recensioni tenendo in considerazione il giorno della settimana cosicché da mettere in risalto pattern di attività.
Nel caso specifico, come è possibile osservare in Figura 2.6, non vi è una dominanza degna di nota nonostante vi sia una tendenza a produrre meno recensioni nelle giornate di venerdì e sabato.

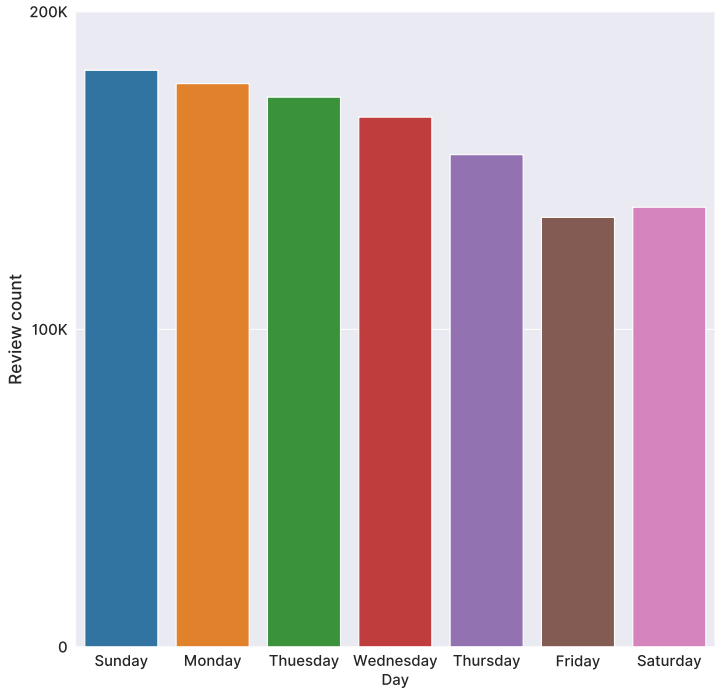


Figure 2.6: Review distribution per day

2.4 Prodotti più recensiti e recensori più popolari

Il numero di utenti e di prodotti è nell'ordine delle migliaia (come anticipato nel Capitolo 2.2) ed è impensabile anche solo immaginare di fare analisi esplorative approfondite su ogni singolo utente e su ogni singolo prodotto. Per questo motivo abbiamo deciso di focalizzare l'attenzione su un numero ristretto di utenti e di prodotti.
La Figura 2.7 mostra i 20 prodotti più popolari in termini di recensioni. Possiamo notare come, seppur ogni prodotto abbia perlopiù un maggior numero di recensioni *positive*, per alcuni prodotti in particolare la percentuale di recensioni *neutrali* e *negative* è elevata rispetto alla distribuzione osservata nel Capitolo 2.3.1.

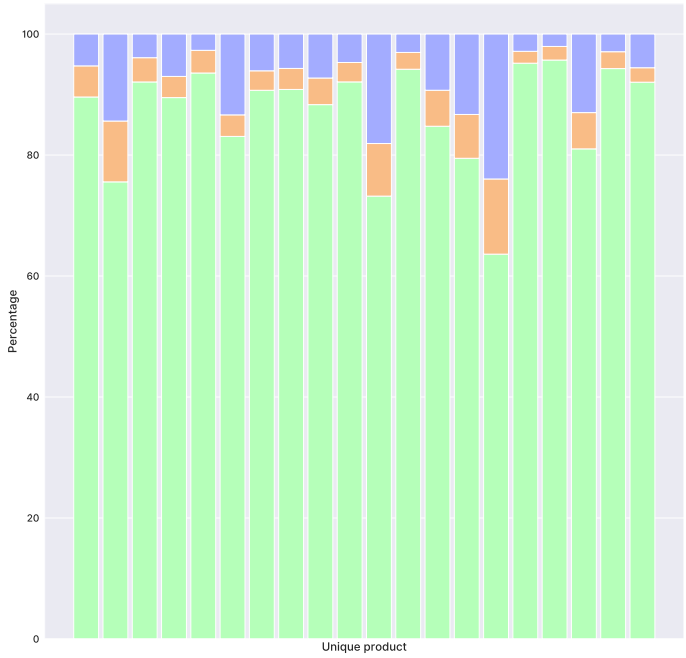


Figure 2.7: Opinion for bestseller products

La Figura 2.8 mostra i 50 utenti con più recensioni prodotte, mentre la Figura 2.9 mostra la distribuzione delle valutazioni delle recensioni effettuate. Possiamo notare come la maggior parte degli utenti considerati dia in percentuale una valutazione in linea con la distribuzione osservata nel Capitolo 2.3.1, fatta eccezione per casi estremi.

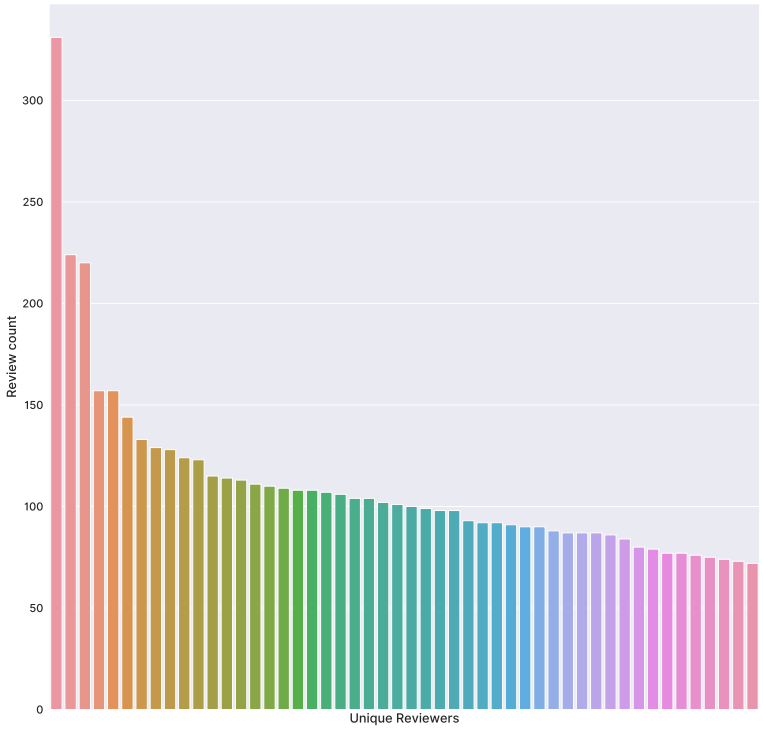


Figure 2.8: Reviewers with most reviews

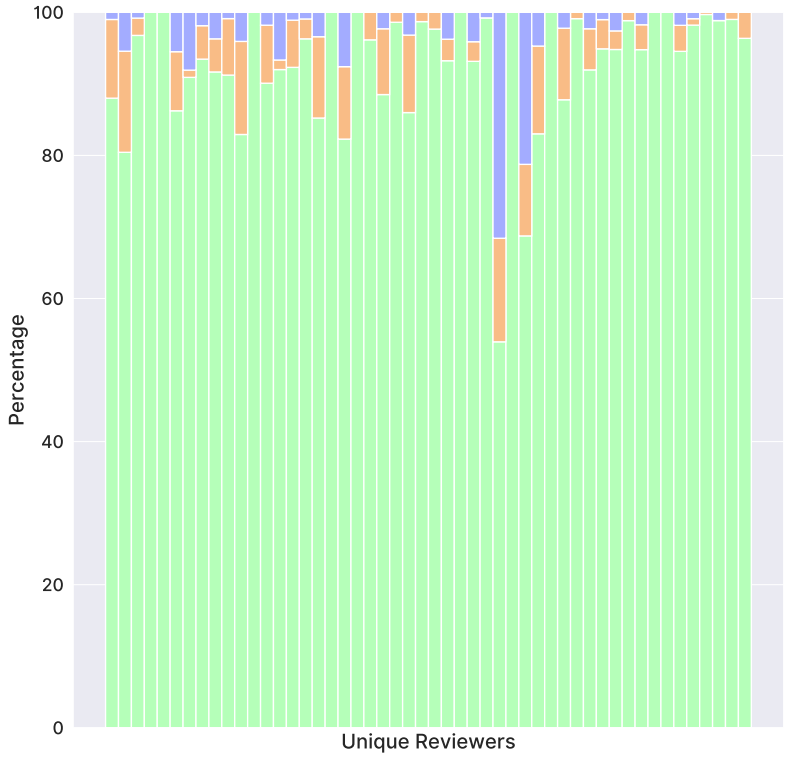
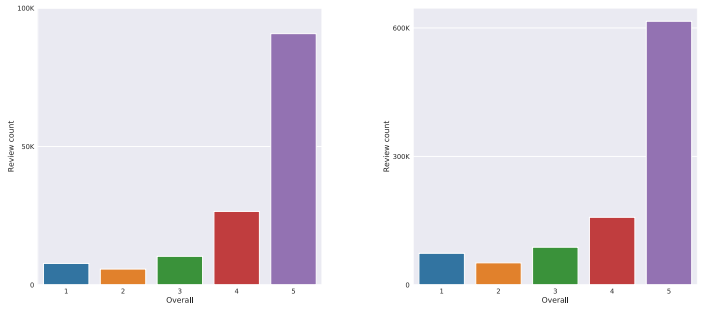


Figure 2.9: Opinion of top reviewers

2.5 Natura delle recensioni

Il campo `verified` merita una trattazione dettagliata per capire se le recensioni `non verificate` sono di valore tanto quanto le recensioni *verificate*. In Figura 2.10 si può notare che la distribuzione del campo *overall* è praticamente identica.



(a) Unverified overall distribution (b) Verified overall distribution

Figure 2.10: Verified - Unverified overall distribution

In Figura 2.11 è invece possibile notare una particolarità. Come in Figura 2.9 abbiamo preso i 50 utenti con più recensioni prodotte e la maggior parte delle loro recensioni risulta come *non verificata*.
Alcune riflessioni sono possibili soffermandoci su questa Figura. Come specificato nel Capitolo 2.2, le recensioni sono classificate come *verificate* se provengono da un acquisto su Amazon per almeno l'80% del valore originale dell'articolo. Una suggestione potrebbe far propendere per l'idea che molti di questi utenti siano i cosiddetti *top recensori* solitamente posizionati in cima alla lista dei commenti che (in teoria) non acquistano direttamente i prodotti recensiti che invece gli vengono prestati per provare il prodotto e scrivere una recensione imparziale.

Figure 2.11: Verified - Unverified reviews of top reviewers

2.6 Correlazioni temporali e relative al traffico

Aggregando temporalmente i dati abbiamo ottenuto alcuni grafici che suggeriscono correlazioni interessanti: 2.12 ci mostra come grande parte del traffico attivo sulle recensioni (ovvero gli utenti che votano e danno rilevanza alle recensioni esistenti) si distribuisce su quelle già più popolari, mentre la grande rimanenza rimane quasi intoccata da grossi picchi di attività di questo tipo.
2.13 mostra un fenomeno interessante: nonostante la quantità di recensioni cambi notevolmente nel tempo, la quantità di recensioni non verificate in rapporto al totale sembra rimanere (quasi) invariata, suggerendo un qualche tipo di moderazione.
Incrociando la lunghezza media delle recensioni con il loro voto, abbiamo ottenuto la figura 2.14. Con il passare del tempo (e l'aumentare vertiginoso del traffico) le recensioni sono generalmente più lunghe e meno generose con la valutazione che esprimono.
Basandosi su alcuni di questi aspetti, Amazon ha sviluppato un modello di apprendimento automatico per assegnare un valore di rilevanza alle recensioni, in modo da poterle mettere in primo piano. In particolare, i fattori considerati sono: punteggio "utilità" della recensioni (voti), recensione verificata/non verificata, età della recensione.
Non è noto nel dettaglio come funzioni e in che modo questi fattori vengano pesati ma è certamente importante rilevare come un approccio di questo tipo permette ad Amazon di sfruttare i contributi degli utenti e capitalizzarci, promuovendo recensioni convincenti e prodotti che riescono a produrre (legittimamente o no, altro aspetto importante) feedback così positivi e virali.

Figure 2.12: Average "helpfulness" of 25 and 200 most relevant reviews over time and traffic

Figure 2.13: Verified - Unverified reviews over time and traffic

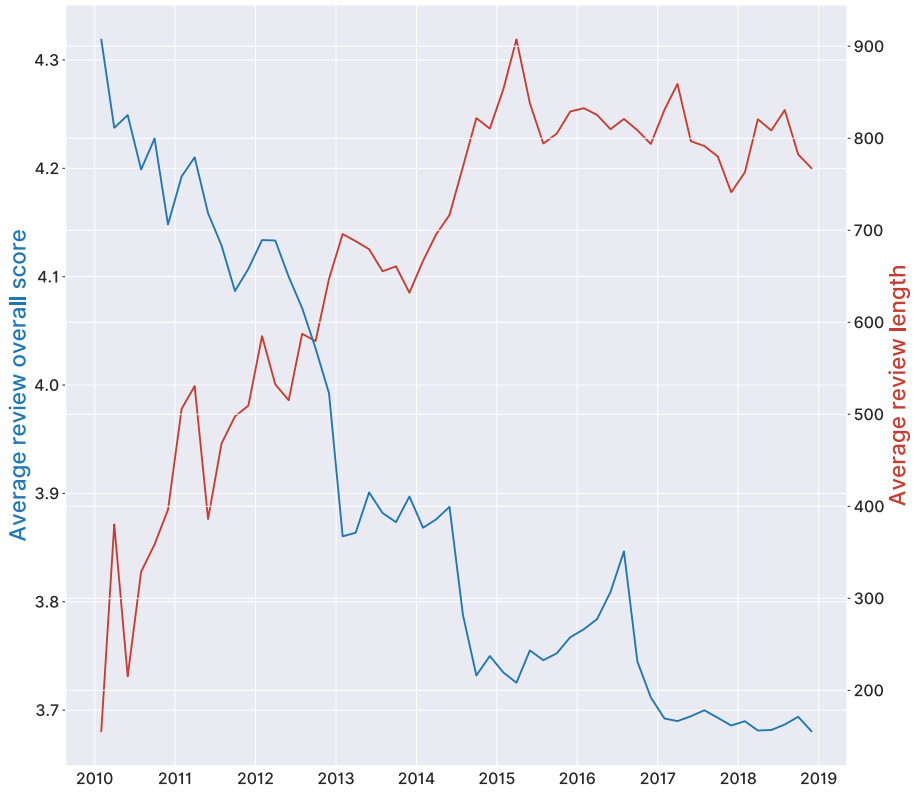


Figure 2.14: Review length VS overall score over time

2.7 Polarizzazione delle valutazioni

Uno degli aspetti fondamentali e poco chiaro delle recensioni è quanto esse siano polarizzate attorno un singolo voto numerico, in modo spesso estremo. È un fenomeno che si estende per ogni categoria di ogni marketplace in modo praticamente omogeneo. Nel caso di Amazon, la gran parte delle recensioni riporta una valutazione numerica massima.
In letteratura, abbiamo trovato un recente lavoro [7] che investiga dettagliatamente la questione, descrivendo la *polarity self-selection* come fattore trainante di questo fenomeno. È tendenza dei consumatori a recensire esperienze estreme. Si discute inoltre il fatto che le distribuzioni estreme di queste valutazioni ne riducono l'informatività, su larga scala.
I seguenti grafici danno un'idea di questo comportamento: la figura 2.15 confronta recensori abituali ed occasionali del sito Yelp, mostrando come utenti che producono più recensioni distribuiscono meglio le proprie valutazioni, senza esagerare con valutazioni massime nella maggior parte dei casi. 2.16 affronta invece l'aspetto dell'incipit della recensione: quando siamo forzati a valutare un elemento, è più probabile che distribuiremo intro al 4 la nostra valutazione, mentre quando lasciamo una recensione di nostra spontanea volontà si tende a recensire ottime esperienze.

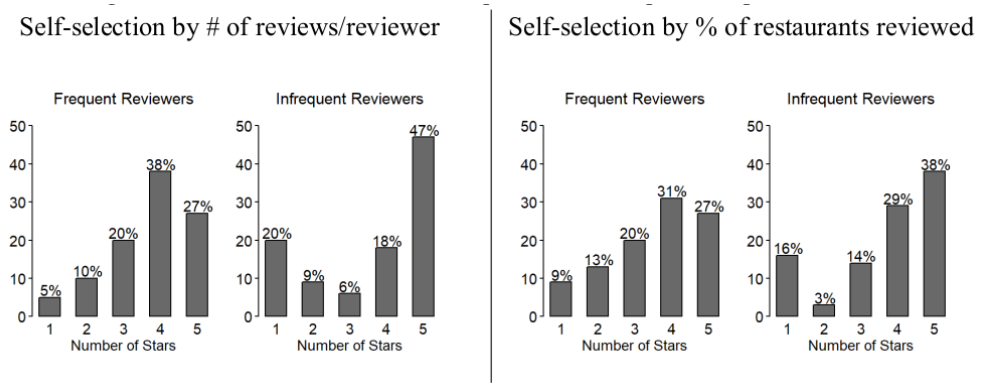


Figure 2.15: Review Distribution of Frequent and Infrequent Yelp Reviewers [7]

Figure 2.16: Empirical Distributions for Self-Selection versus Forced Reviews [7]

Chapter 3

Sentiment analysis

Oggigiorno siamo affetti da e produciamo un tale sovraccarico di dati che le aziende si stanno ridefinendo per raccogliere queste informazioni, come per esempio i feedback dei clienti, e strutturare il processo decisionale. L'ottenimento di questi dati è impensabile se fatto manualmente.
In particolare, per le opinioni su prodotti e servizi viene in aiuto la sentiment analysis, una discplina che può fornire risposte riguardo le questioni più importanti dal punto di vista dei clienti.
Il processo di sentiment analysis permette, attraverso l'elaborazione del linguaggio naturale, di estrarre e analizzare in modo automatizzato opinioni soggettive espresse dall'utente, determinarne la polarità (positiva, neutrale, negativa) e, successivamente, riassumerle in maniera da poter essere di valore per l'azienda. In questo modo, le decisioni possono essere prese sulla base di una quantità di dati significativa, piuttosto che da una semplice intuizione che non sempre si rivela corretta. Il rischio infatti a cui si va incontro maggiormente è quello di interpretare i messaggi avendo già un pregiudizio sull'argomento in questione, influenzando il modo in cui il testo da analizzare può essere interpretato.
La sentiment analysis è importante perché le aziende vogliono che il loro marchio sia recepito positivamente (con un occhio alle aziende concorrenti). A tal proposito, ci si può concentrare su commenti positivi o negativi oltre che sul feedback del cliente, per valutare sia i punti di forza che i punti su cui migliorare.

3.1 Preprocessing

Prima di partire con lo svolgimento del task di sentiment analysis, è necessaria una fase di preprocessing.
Innanzitutto, sono stati rimossi dal dataset i campi ritenuti superflui per l'analisi. Successivamente, è stato manipolato il campo `reviewText`. La manipolazione è avvenuta sequenzialmente e con step standard per analisi di questo tipo:

- **Normalization** - conversione delle recensioni in caratteri minuscoli. Se presenti, modificate alcune espressioni contratte tipiche della lingua inglese (per esempio: *hadn't* trasformata in *had not*).
- **Tokenization** - suddivisione in token per ogni recensione
- **Removal** - rimozione di token altamente ricorrenti nella lingua considerata (stopwords). Inoltre, sono stati eliminati token composti da 1 o 2 caratteri o token estremamente rari (frequenza nel dataset = 1)
- **Lemmatization** - conversione del token nel proprio lemma linguistico

Questa manipolazione ha portato alla creazione del campo `preprocessedReview`. Di seguito vengono mostrate la recensione originale e la recensione dopo l'intera fase di preprocessing.

Prima `Overall a great product with a fair price. I have had absolutely no problems with the product except for the volume level, which is *NOT* below standard, it is just simply what is to be expected from a headset. Very comfortable, and I personally prefer the boom mic to be longer (unlike the newer models of this headset which have shortened mics). Recommended.`

Dopo `overall great product fair price absolutely problem product volume level standard simply expect headset comfortable personally prefer boom mic longer new model shorten mic recommend`

3.2 Creazione di Bag of Words

Il campo `preprocessedReview` non è direttamente trattabile dagli algoritmi di machine learning e quindi è necessario ottenere una rappresentazione comprensibile. Innanzitutto, abbiamo rimosso dall'analisi del campo `preprocessedReview` le recensioni:

- prolisse - rientrano in questa categoria le osservazioni con più di 300 parole
- irrilevanti - rientrano in questa categoria le osservazioni con meno di 5 parole

Dopodiché, le recensioni restanti sono state utilizzate per costruire una Bag of Words composta da 10000 feature. Oltre ai token vengono considerati anche i bigrammi, poiché un loro utilizzo può aumentare l'accuratezza del modello rispetto al solo utilizzo di token singoli.

3.3 Esplorazione

A partire dalla nuova rappresentazione matriciale è stato creato un DataFrame fittizio composto dalle 10000 feature individuate nel Capitolo 3.2 e, per ognuna

di esse, viene segnata la frequenza con cui appaiono rispettivamente nelle recensioni positive e nelle recensioni negative, oltre che la frequenza totale data dalla loro somma.
Le recensioni neutrali non sono state considerate in quanto aggiungerebbero un livello ulteriore di complessità nell'apprendimento dei modelli.

3.3.1 Wordcloud

Una Wordcloud è una rappresentazione grafica delle parole usate di frequente in un corpus di documenti e fornisce un'idea generale di che tipologia di parole possiamo trovarvi. La grandezza di ogni parola nell'immagine è un'indicazione della frequenza di occorrenza della parola nell'intero corpus. Per questo motivo, sono molto utili quando si vuole eseguire un'analisi del testo.

Figure 3.1: Wordcloud of positive reviews

Figure 3.2: Wordcloud of negative reviews

3.3.2 Frequenza dei token

Un'ulteriore analisi che si può fare riguarda la distribuzione dei token all'interno del dataset. Nella Figura 3.3 e nella Figura 3.4 vengono mostrati gli istogrammi (ordinati dal token più frequente al token meno frequente) rispettivamente per le recensioni positive e le recensioni negative.

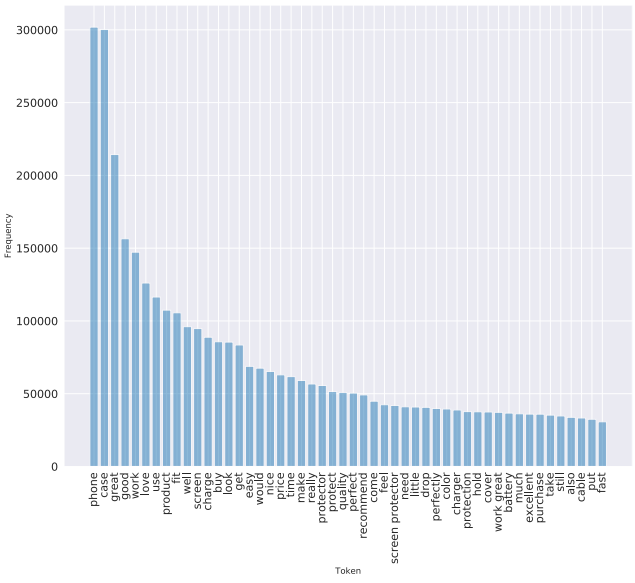


Figure 3.3: Top 50 tokens in positive reviews

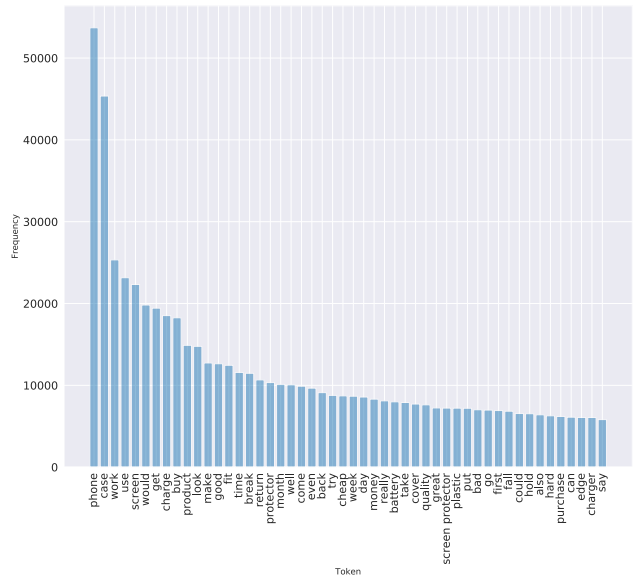


Figure 3.4: Top 50 tokens in negative reviews

Un modello comunemente utilizzato è la legge di Zipf, ovvero una legge empirica formulata nel 1959 in cui vi si afferma che, dato un corpus di documenti, la frequenza di ogni parola è inversamente proporzionale al suo rango nella tabella delle frequenze. Pertanto, la parola più frequente ricorre approssimativamente il doppio rispetto alla seconda parola più frequente, il triplo rispetto alla terza parola più frequente e così via.

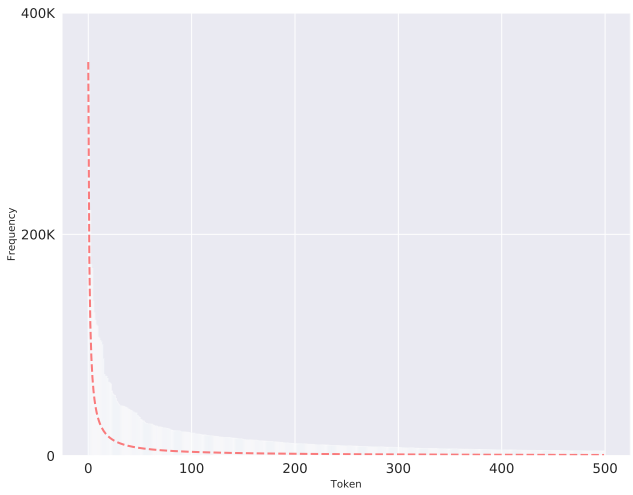


Figure 3.5: Distribution of words in review for each opinion

La legge di Zipf viene osservata più facilmente tracciando i dati su scala logaritmica in entrambi gli assi come mostrato in Figura 5.3.

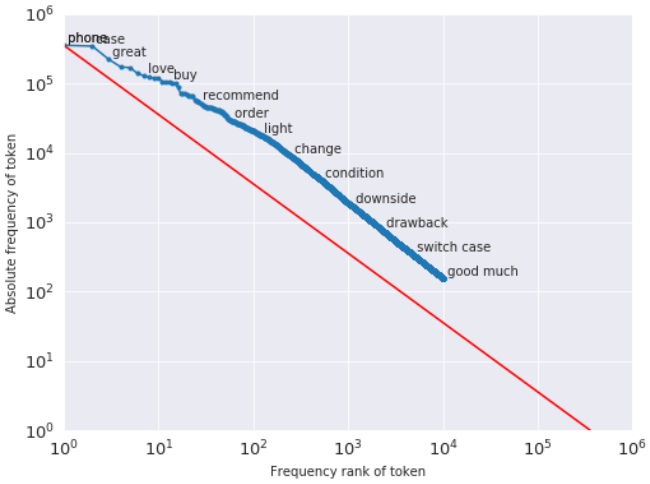


Figure 3.6: Distribution of words in verified reviews

3.4 Machine learning

La tecnica con cui abbiamo affrontato la fase di sentiment analysis è il machine learning. Grazie ad essa il modello viene addestrato per riconoscere il sentimento in base alle parole usando un training set etichettato. Questo approccio dipende in gran parte dal tipo di algoritmo e dalla qualità dei dati utilizzati per l'addestramento.
Grazie all'attenta fase di preprocessing, è stato possibile addestrare due modelli diversi con gli stessi dati e valutarne i risultati, confrontandoli.
Per prima cosa abbiamo dovuto risolvere il problema dello sbilanciamento tra classi: le recensioni positive sono in numero molto maggiore rispetto a quelle negative e qualsiasi modello addestrato rischierebbe di ottimizzarsi più sulla classe maggiore. La soluzione più adeguata in questo caso è l'utilizzo di una tecnica di undersampling in modo da ridurre gli elementi della classe maggioritaria senza introdurre bias e rendendo equiparabili le cardinalità delle classi.
Il problema di machine learning è di natura binaria in quanto la variabile target ha solo due possibili valori: *positive* e *negative*. Data questa semplicità ci sono numerosi modelli che possono essere addestrati. Sono stati scelti Multinomial Naive Bayes e Logistic Regression per il loro ottimo compromesso tra performance e tempo di addestramento. Abbiamo provato ad implementare anche una Support Vector Machine ma le risorse hardware richieste per l'addestramento non erano disponibili.
Per entrambi i modelli è stata eseguita la tecnica di Cross Validation (CV) su 5 folds, dividendo quindi il training set in 5 sottoinsiemi e usandone uno come test set, per 5 volte.

to do:split da aggiungere Per entrambi i modelli è stato tenuto da parte un validation set per la verifica finale e l'analisi delle varie metriche.

3.4.1 Analisi dei risultati

Al di sopra della CV è stata eseguita anche una Grid Search nel tentativo di trovare i migliori iperparametri per i due modelli. In Naive Bayes è stato trovato il migliore valore per alpha (0.1), mentre nella Logistic Regression il migliore valore dell'iperparametro C è stato 1.
Analizzando le matrici di confusione dei due modelli in Figura 3.7 e in Figura 3.8 notiamo che il modello di Logistic Regression individua in totale meno falsi positivi e falsi negativi rispetto a Naive Bayes; ciò è confermato dal valore di *Accuracy* leggermente più alto.

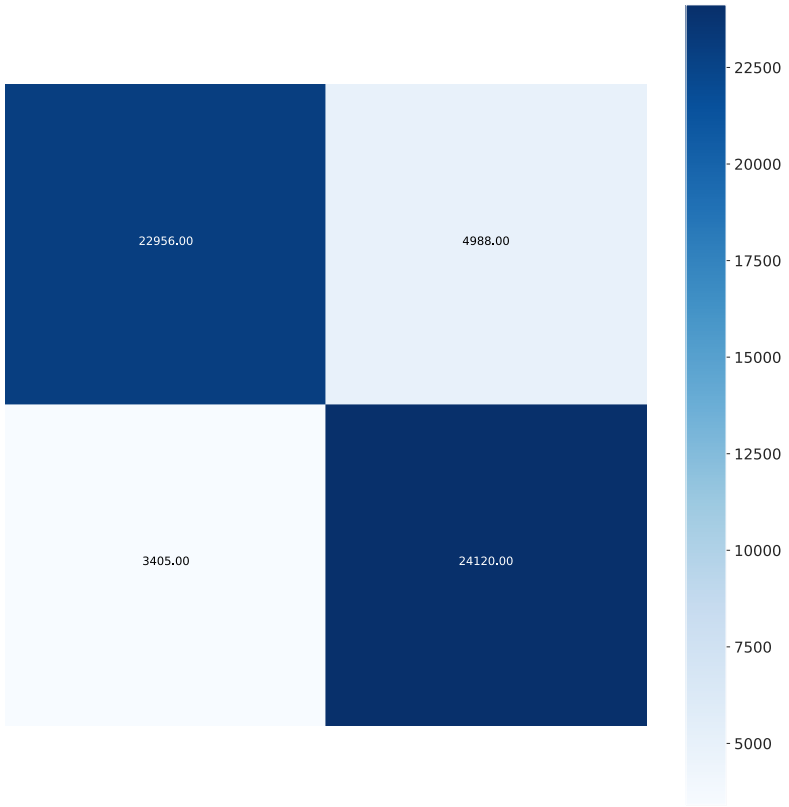


Figure 3.7: Confusion Matrix per Naive Bayes

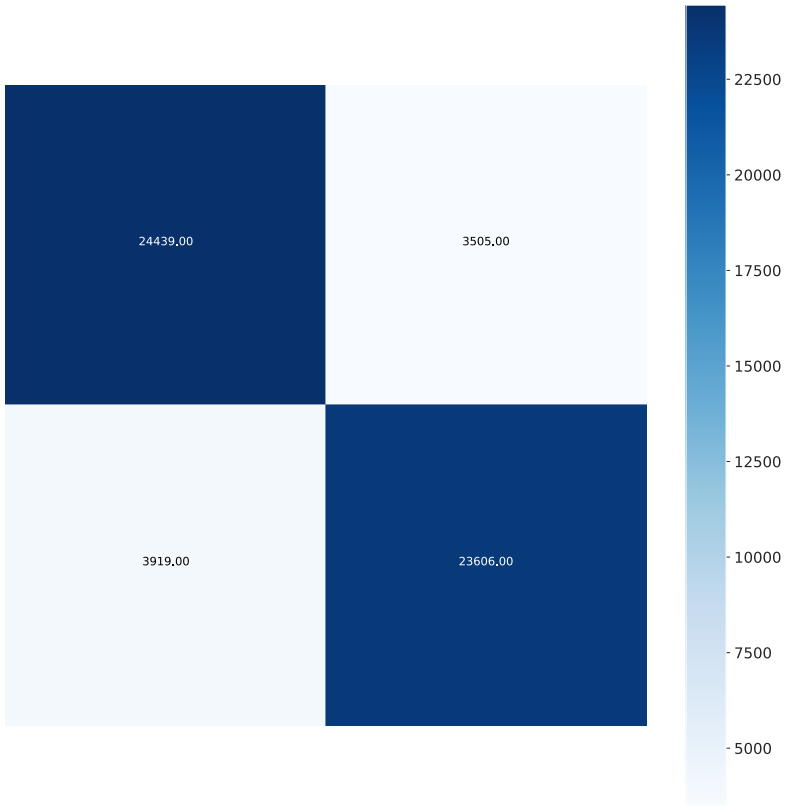


Figure 3.8: Confusion Matrix per Logistic Regression

Sia i valori di *Precision* che quelli di *Recall* di Logistic Regression sono più alti indicando che per entrambe le classi tale modello trova un miglior numero di True e un minor numero di False.

Accuracy complessiva	0.84
Precision per la classe *positive*	0.84
Precision per la classe *negative*	0.85
Recall per la classe *positive*	0.85
Recall per la classe *negative*	0.84

Table 3.1: Metriche risultate dell'esecuzione della cross validation su Naive Bayes

Accuracy complessiva	0.87
Precision per la classe *positive*	0.87
Precision per la classe *negative*	0.86
Recall per la classe *positive*	0.86
Recall per la classe *negative*	0.88

Table 3.2: Metriche risultate dell'esecuzione della cross validation su Logistic Regression

Le due ROC in Figura 3.9 e in Figura 3.10 mostrano l'andamento del rapporto tra True Positive Rate e False Positive Rate al variare del valore di cut-off di ogni modello: anche in questo grafico possiamo vedere una performance migliore da parte della Logistic Regression poichè la sua curva si avvicina di più a quella "ideale" e la sua inclinazione è più verticale.

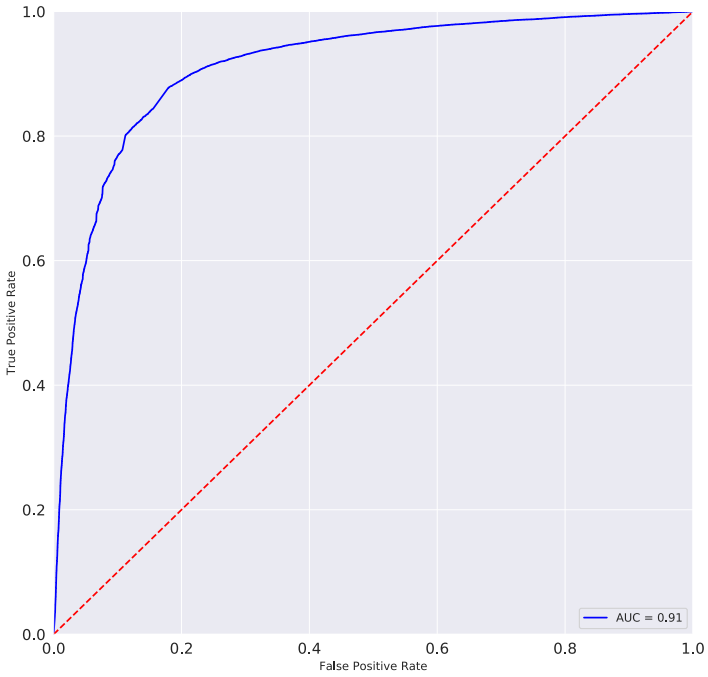


Figure 3.9: ROC per Naive Bayes

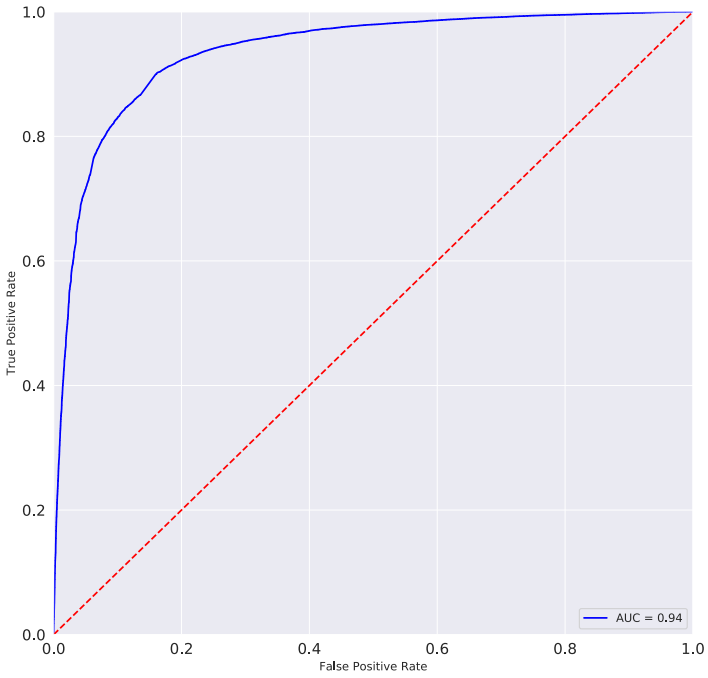


Figure 3.10: ROC per Logistic Regression

Chapter 4

Topic analysis

La topic analysis consente di identificare gli argomenti più discussi semplicemente contando le parole all'interno di un corpus di documenti e raggruppando modelli di parole simili.
È una tecnica di machine learning che organizza e comprende grandi raccolte di dati testuali, assegnando tag o categorie in base all'argomento o al tema di ogni singolo testo.
I risultati sono più dettagliati e interessanti rispetto alla sentiment analysis, in quanto la topic analysis esamina più da vicino le informazioni dietro un testo.
Sono comunque due metodi che, se usati in combinazione, consentono di restringere ulteriormente queste informazioni per trovare con precisione quali temi vengono discussi, fornendo quindi informazioni fruibili riguardanti il prodotto.

4.1 Algoritmo utilizzato

Il metodo di riferimento di topic analysis è Latent Dirichlet Allocation (LDA) [2]: è una tecnica di machine learning non supervisionata che consente di inferire schemi e raggruppare espressioni simili senza la necessità di definire gli argomenti a priori. L'assunzione di LDA è che ogni documento può essere descritto da una distribuzione di argomenti, e ciascun argomento può altresì essere descritto da una distribuzione di parole.
LDA è un modello ampiamente documentato in Python e per questo motivo è stata la scelta naturale per effettuare il task di topic analysis.
Detto questo, è da far notare che la letteratura scientifica negli ultimi anni ha prodotto diversi modelli che mettono in risalto alcuni svantaggi del modello LDA.

4.1.1 Individuazione topic

Una limitazione è stata riscontrata nello sviluppo del modello MG-LDA in cui viene asserito che i modelli standard (come LDA) tendono a produrre topic che

corrispondono alle proprietà globali degli oggetti in analisi piuttosto che agli aspetti di un oggetto che tendono ad essere valutati da un utente.
La soluzione adottata in questo progetto, vista l'assenza di una implementazione per il modello MG-LDA [3], è stata quella di applicare LDA su prodotti presi singolarmente - considerando ovviamente i prodotti con più osservazioni all'interno del dataset - nonostante la pratica più diffusa in letteratura sembra quella di applicare il modello sull'intero corpus di documenti a prescindere dall'eterogeneità dei documenti stessi. Questa scelta è stata presa di proprosito per evitare la formazione di macro-topic.

4.1.2 Sentiment topic

Un'altra carenza riscontrata è l'assenza di rilevazione del sentiment: questo compito è risolto da diversi modelli (per esempio JST [4], basato su LDA), che suddividono il testo in argomenti, assegnando simultaneamente a ciascuno un livello di sentimento.
L'idea per questo progetto, vista la mancanza di una implementazione del modello JST, è stata quella di utilizzare un approccio lexicon-based (VADER [5]) in combinazione con l'approccio non supervisionato, in modo da poter fornire una visione generale ed approssimata del sentiment dei topic prodotti da LDA. Di seguito viene presentata la procedura implementata:

- Per ogni recensione viene calcolato il rispettivo sentiment. In particolare, l'output fornito è una variabile `compound` $\in [-1, +1]$ che rappresenta un sentiment:
  - **positivo** se `compound` ≥ 0.05
  - **neutrale** se -0.05 $<$ `compound` < 0.05
  - **negativo** se `compound` ≤ -0.05
- In seguito all'applicazione del modello LDA, ogni recensione ha associata la probabilità con la quale ha contribuito alla formazione del topic.
- Per ogni recensione viene estratto il topic con probabilità maggiore (se $>$ 0.7). In questo caso, si assume che la recensione sia inerente al suddetto topic, altrimenti viene scartata dal conto.
- Per ogni topic viene effettuata la media del sentiment delle recensioni associate al suddetto topic

Questa soluzione non è parsa consistente in quanto circa il 0.1% delle recensioni appartenevano ad un topic specifico con probabilità superiore a 0.7. L'abbassamento della soglia di probabilità è anch'essa una soluzione inconsistente.

4.2 Procedimento

La fase di topic analysis condivide le operazioni preliminari di preparazione del dataset presentate nel Capitolo 3.1 e nel Capitolo 3.2.

L'algoritmo LDA è stato applicato su oggetti presi singolarmente per i problemi sottolineati nel Capitolo 4.1.1. Il criterio di scelta dei prodotti da analizzare è la loro popolarità in termini di recensioni.
In particolare, soffermandoci sulla Figura 2.7 si può notare quanto sono dominanti, in percentuale, le recensioni positive. Ciò rispecchia connotati già riscontrati durante l'esplorazione del dataset; nonostante ciò alcuni prodotti tra i più recensiti mostrano una percentuale di recensioni negative e neutrali elevata rispetto alla media.
Considerando i tempi di computazione, un totale di 6 prodotti si è rivelato essere un buon trade-off per un'analisi variegata, prendendo i 3 prodotti con la media di `overall` più alta e i 3 prodotti con la media di `overall` più bassa.
Essendo LDA un algoritmo non supervisionato che produce argomenti astratti senza conoscerne il numero a priori, solitamente necessita di un tuning degli iperparametri per individuare il modello migliore. Gli iperparametri sono:

- K: è il numero di argomenti da estrarre dal corpus di documenti disponibile
- α: è il parametro relativo alla distribuzione che regola l'aspetto della distribuzione degli argomenti per tutti i documenti del corpus. Tipicamente viene scelto un valore di $\alpha < 1$ per ottenere una distribuzione sparsa di argomenti per documento.
- β: è il parametro relativo alla distribuzione che regola l'aspetto della distribuzione delle parole in ciascun argomento. Per lo stesso motivo di α, viene scelto un valore $\beta < 1$.

Nella Tabella 4.1 vengono mostrati i possibili valori assumibili dagli iperparametri.

Iperparametro	Valori possibili
K	[2, 3, 4, 5, 6, 7, 8, 9, 10]
α	[0.1, 1]
β	[0.01, 0.1, 1]

Table 4.1: Possibili valori degli iperparametri di LDA

Per valutare la qualità degli argomenti appresi viene usato il punteggio di *coherence*. Per ogni prodotto:

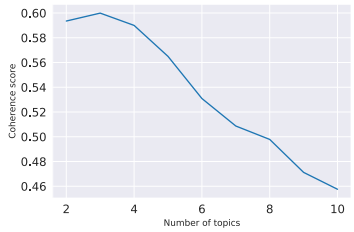
- Viene applicato l'algoritmo iterando sull'insieme di iperparametri
- Ogni modello risultante ottiene un punteggio
- Il modello con il punteggio più alto è l'ottimale

Nella Tabella 4.2 vengono mostrati i modelli ottimali per ciascun prodotto considerato per l'analisi.

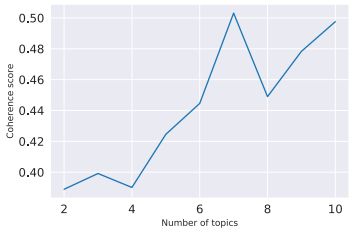
Codice prodotto	α	β	K	Coherence score
B00MXWFUQC	1	1	3	0.48
B0092KJ9BU	0.1	1	7	0.50
B00UC7G565	0.1	0.01	2	0.55
B00VH88CJ0	0.1	0.01	2	0.57
B005NF5NTK	1	0.1	3	0.60
B00X5RV14Y	0.1	1	6	0.55

Table 4.2: Iperparametri del modello ottimale con rispettivo punteggio di coherence

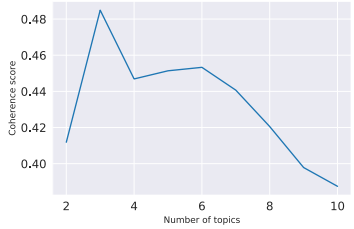
In Figura 4.1 vengono invece mostrati i grafici del punteggio di *coherence* per ciascun prodotto e con gli iperparametri α e β mostrati nella Tabella 4.2.



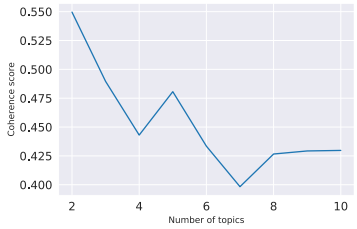
(a) Coherence plot of B005NF5NTK



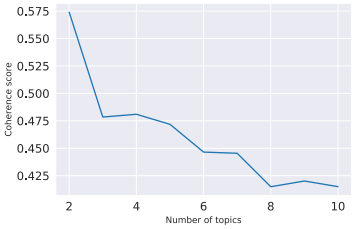
(b) Coherence plot of B0092KJ9BU



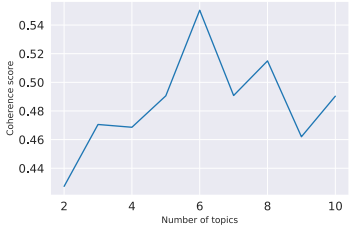
(c) Coherence plot of B00MXWFUQC



(d) Coherence plot of B00UCZGS6S



(e) Coherence plot of B00VH88CJ0



(f) Coherence plot of B00X5RV14Y

Figure 4.1: Coherence plots of products

4.3 Visualizzazione dei risultati

Per valutare i risultati prodotti dall'algoritmo LDA abbiamo usufruito di pyLDAvis, una libreria Python basata su [6] che permette di visualizzare gli argomenti in maniera interattiva.
Fornisce una visione globale degli argomenti e di come differiscono l'uno dall'altro, consentendo allo stesso tempo un'analisi approfondita dei termini maggiormente associati a ciascun singolo argomento.
Il pannello di sinistra visualizza gli argomenti come cerchi nel piano bidimensionale i cui centri sono determinati calcolando la divergenza di Jensen-Shannon tra gli argomenti.
Il pannello di destra mostra un grafico a barre orizzontali, le cui barre rappresentano i (30) singoli termini che sono i più rilevanti per interpretare l'argomento attualmente selezionato a sinistra. Una coppia di barre sovrapposte rappresenta sia la frequenza di un determinato termine a livello di corpus (barre blu) sia la frequenza specifica dell'argomento del termine (barre rosse).
Sempre nel pannello di destra, appena sopra il grafico a barre orizzontali, è possibile regolare per mezzo di uno slider il valore λ, con $0 \leq \lambda \leq 1$. Esso consente di classificare la pertinenza di un termine rispetto a un argomento.
Valori di λ vicino a 0 evidenziano i termini potenzialmente rari ma esclusivi per l'argomento selezionato, mentre valori di λ vicino a 1 evidenziano i termini frequenti ma non necessariamente esclusivi per l'argomento selezionato.
L'impostazione consigliata in [6] suggerisce un valore di λ intorno a 0.6, che è stato dimostrato essere di aiuto per gli utenti per interpretare l'argomento, nonostante sia fatto presente che il valore ottimale può variare in base al dataset e gli argomenti stessi.

Chapter 5

Web app

Abbiamo sviluppato una applicazione web interattiva che dimostra alcuni dei nostri risultati.
È strutturata in due parti secondo i principi Restful: Il backend utilizza Flask per poter offrire una semplice API attraverso il quale è possibile sfruttare i modelli allenati ed esporne la funzione `pred_prob`, in modo da visualizzare in tempo reale il comportamento del classificatore su di un testo personalizzato, non facente parte del dataset iniziale. Il sistema che abbiamo costruito per esporre i modelli fittati è un buon modo di rendere utilizzabile da chiunque, senza dover passare da script, i risultati del nostro progetto. È facilmente estendibile per altri classificatori e parametri.
Il frontend è sviluppato con Vue JS, un framework Javascript per sviluppare applicazioni reattive. Offre un'interfaccia che consuma l'API appena descritta, visualizzando il risultato della computazione.
Per LDA, un'altra pagina nella stessa applicazione raccoglie sei plot interattivi generati da pyLDAvis, permettendo di consularli e mostrando descrizione, codice e titolo di ognuno degli articoli a cui si riferiscono.

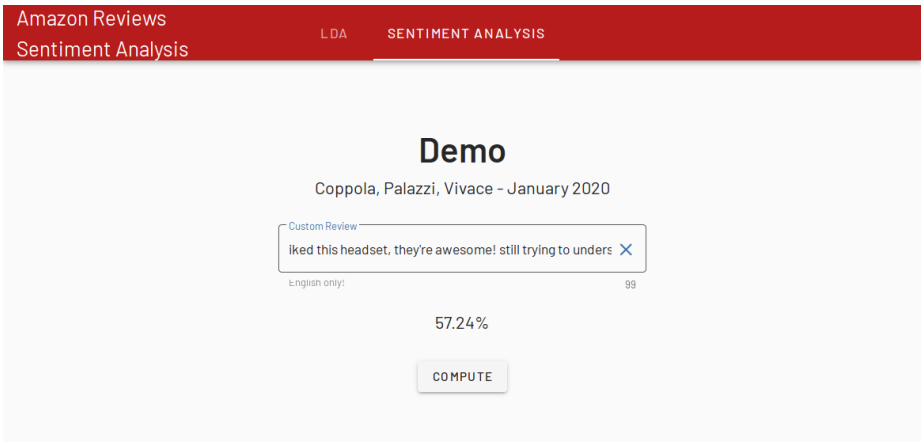


Figure 5.1: Vista Sentiment Analysis della Demo

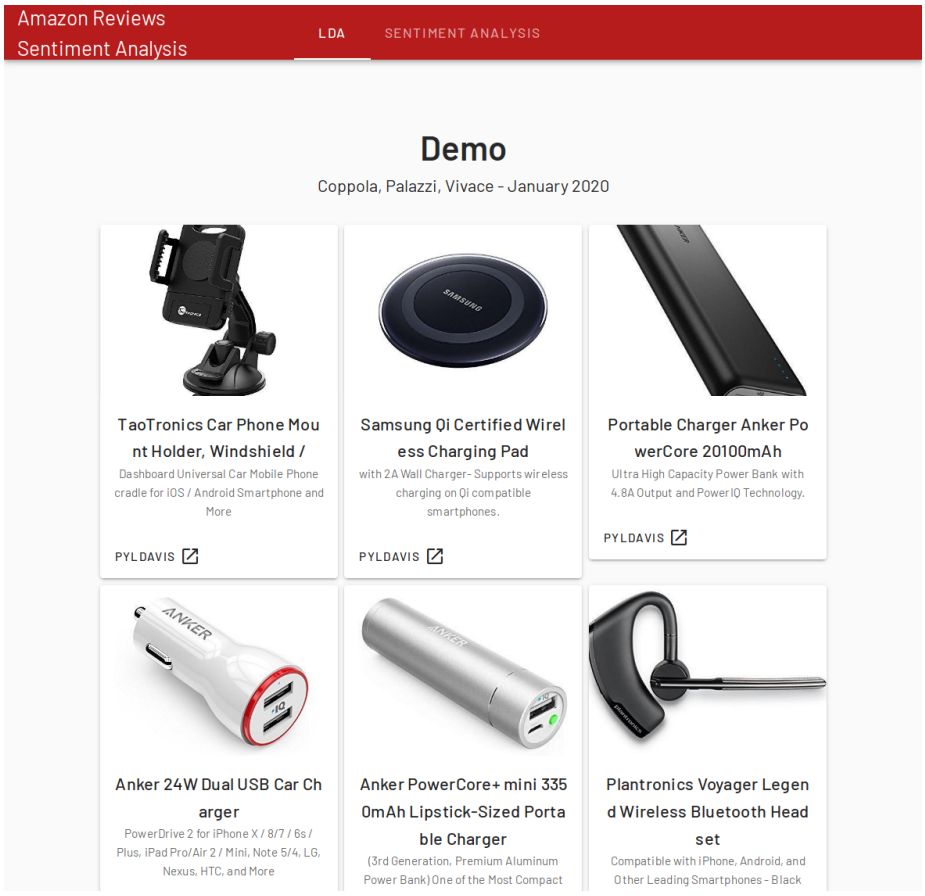


Figure 5.2: Vista LDA della Demo (pyLDAvis)

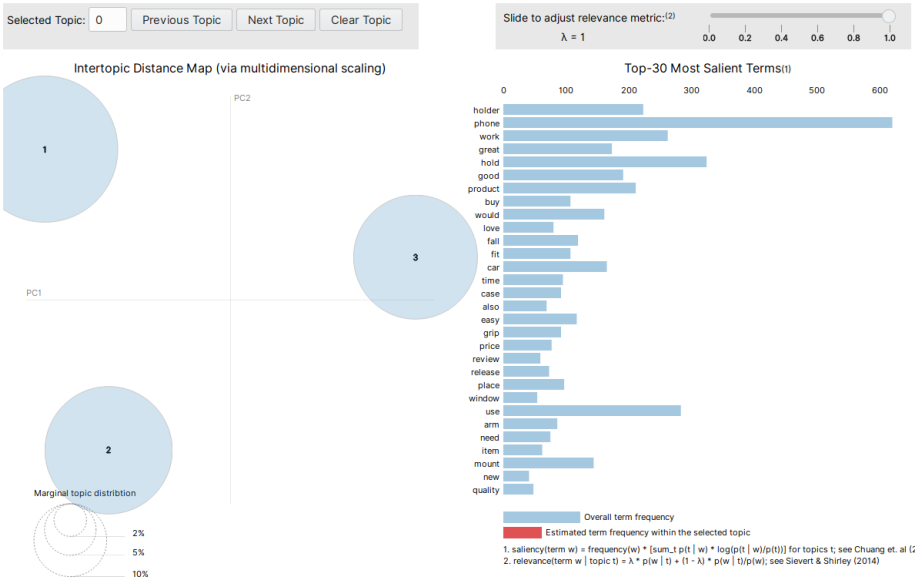


Figure 5.3: Export pyLDAvis per un singolo prodotto

Chapter 6

Conclusioni

L'esplorazione delle recensioni di prodotti Amazon ci ha permesso di constatare l'enorme numero di informazioni che possono essere estratte da opinioni degli acquirenti con lo scopo di stilare statistiche e valutazioni e poter quindi prendere decisioni in ambito aziendale per migliorare i servizi offerti o centrare meglio la propria clientela.
Lo studio di sentiment analysis dimostra che si possono ottenere modelli addestrati con metriche di controllo molto soddisfacenti e pronti per essere usati nell'analisi sentimentale delle future recensioni.
Per quanto concerne la topic analysis, gli argomenti individuati sui prodotti più recensiti attraverso il modello LDA non sempre sono facili da interpretare. Abbiamo evitato la creazione di argomenti troppo generali ma non abbiamo sempre ottenuto argomenti facilmente utilizzabili per fare ragionamenti complessi sui prodotti. Con ogni probabilità, strumenti più avanzati di Topic-Sentiment Analysis porterebbero ad una scelta degli argomenti più logica ed intuitiva.
I modelli utilizzati sono facilmente adattabili a qualsiasi categoria di e-Commerce dotata di una forma di recensioni e, considerati i risultati raggiunti con strumenti base di Data Analytics, non c'è da stupirsi se Amazon sia riuscito a raggiungere la vetta facendo leva su queste tecnologie.

Bibliography

[1] US Census Bureau News. *QUARTERLY RETAIL E-COMMERCE SALES*. https://www.census.gov/retail/mrts/www/data/pdf/ec_current.pdf. 3rd Quarter 2019.

[2] David M Blei, Andrew Y Ng, and Michael I Jordan. "Latent dirichlet allocation". In: *Journal of machine Learning research* 3.Jan (2003), pp. 993–1022.

[3] Ivan Titov and Ryan McDonald. "A joint model of text and aspect ratings for sentiment summarization". In: *proceedings of ACL-08: HLT*. 2008, pp. 308–316.

[4] Chenghua Lin and Yulan He. "Joint sentiment/topic model for sentiment analysis". In: *Proceedings of the 18th ACM conference on Information and knowledge management*. ACM. 2009, pp. 375–384.

[5] Clayton J Hutto and Eric Gilbert. "Vader: A parsimonious rule-based model for sentiment analysis of social media text". In: *Eighth international AAAI conference on weblogs and social media*. 2014.

[6] Carson Sievert and Kenneth Shirley. "LDAvis: A method for visualizing and interpreting topics". In: *Proceedings of the workshop on interactive language learning, visualization, and interfaces*. 2014, pp. 63–70.

[7] Verena Schoenmüller, Oded Netzer, and Florian Stahl. "The extreme distribution of online reviews: Prevalence, drivers and implications". In: *Columbia Business School Research Paper* 18-10 (2018).

[8] Jianmo Ni Amazon. *Amazon Review Data, 2018*. https://nijianmo.github.io/amazon/index.html.

[9] Uma Gajendragadkar. *Product Recommender using Amazon Review dataset*. https://towardsdatascience.com/product-recommender-using-amazon-review-dataset-e69d479d81dd.

[10] Edison Trends. *eBay and Amazon seles trends*. https://trends.edison.tech/research/2018-ebay-vs-amazon.html.