Text mining and search

AMAZON FINE FOOD REVIEWS: CLUSTERING AND CLASSIFICATION

Paparella Matteo 812561 Msc Data Science
Leuce Francesco Mcs Data Science
Zottola Gabriele 812363 Mcs Data Science

June 10, 2021

Contents

1 Introduzione

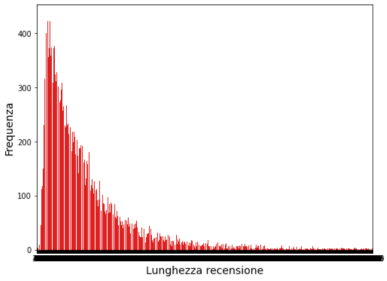
Il progetto ha come oggetto di analisi un dataset di 568.454 recensioni Amazon, prese nel periodo tra Ottobre 1999 e Ottobre 2012, riguardanti 74.258 prodotti alimentari.
Durante lo studio, prendendo in considerazione i testi delle recensioni, lo score assegnato ed applicando differenti tecniche di rielaborazione e rappresentazione testuale, si sono effettuate le seguenti analisi:
- supervised learning and classification in tre differenti classi rappresentanti le recensioni positive, negative e neutre
- supervised learning and classification in cinque differenti classi, rappresentanti il numero di stelle assegnate alla recensione
- unsupervised learning
- sentiment analysis per la classificazione in recensioni positive, negative, neutre e per score
Nel report verranno presentate le tecniche adottate, i risultati ottenuti e le conclusioni tratte dalle analisi effettuate.

1.1 Report Goal

Lo studio si pone come obiettivo la ricerca del metodo di classificazione più efficace, applicando differenti tecniche di preprocessing e rappresentazione testuale.

2 Data Exploration

Il dataset, con 568.454 istanze, presenta i seguenti attributi:
- Id
- ProductID: attributo identificatore del prodotto
- UserID: attributo identificatore dell'utente
- ProfileName: nome di profilo dell'utente
- HelpfulnessNumerator: numero di utenti che hanno trovato utile la recensione
- HelpfulnessDenominator: Numero di utenti che hanno indicato se hanno trovato utile o meno la recensione
- Score: punteggio che va da 1 a 5
- Time: data della recensione
- Summary: breve sommario della recensione
- Text: testo della recensione
Si procede a rilevare il numero di valori nulli presenti nel dataset che si attestano essere unicamente 43, distribuiti nella feauture "ProfileName" (16 valori mancanti) e "Summary" (27 valori mancanti).
Osservando le statistiche descrittive, si nota una media dello Score pari a 4.18, deducendo quindi che le recensioni non siano equamente disribuite per il valore assegnato ad esse.
Si mostra inoltre graficamente la lunghezza dei testi presenti nel dataset.



3 Data Preprocessing

3.1 Eliminazione duplicati

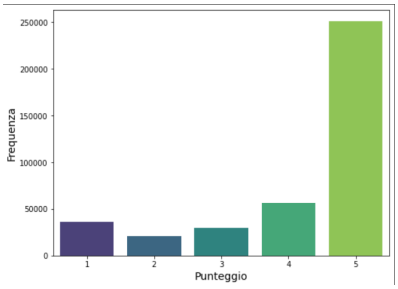
Al fine di ottenere un dataset con recensioni univoche, si eliminano i duplicati presenti, prendendo in considerazione gli attributi "UserId", "ProfileName","Time" e "Text". Il processo di rimozione porta ad una riduzione della dimensione da 568.454 a 393.933 record.

3.2 Feature selection

Dalle 10 feature iniziali si selezionano unicamente il testo della recensione ("Text") e l'attributo "Score".

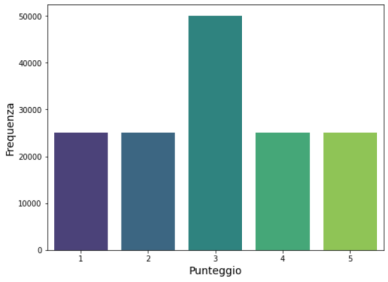
3.3 Balancing

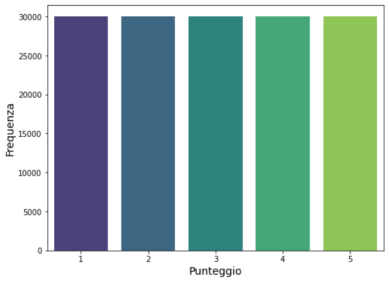
Dall'osservazione preliminare delle statistiche descrittive relative all'attributo "Score" si era dedotta una distribuzione inequa all'interno del dataset. Si decide, dopo la rimozione dei duplicati, di osservare graficamente i valori presenti in modo tale da verficiare eventuali cambiamenti.



Per poter avere un dataset non distorto si è adotta una tecnica di campionamento stratificato, bilanciando il dataframe come segue:
- nel caso di classificaizone in 3 classi si decide di estrarre 50.000 recensioni con score negativo (25.000 score pari ad 1 e 25.000 score pari a 2), 50.000 recensioni neutre (score pari a 3) e 50.000 recensioni positive (25.000 recensioni con score

pari a 4 e 25.000 con score pari a 5)
- nel caso di classificazione in 5 classi si decide di estrarre 5 sample, uno per score, da 30.000 recensioni l'uno
Graficamente si ottengono i seguenti risultati:



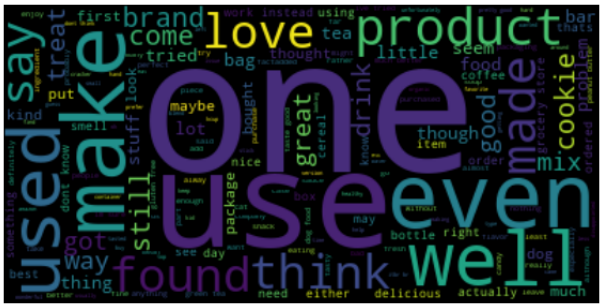


Si riduce quindi ulteriormente la dimensionalità del dataset, andando però ad ottenere un numero di righe che permettono il raggiungimento di un volume considerevole di istanze per il training degli algoritmi di classificaizone supervisionata.

4 Text preprocessing

Prima di adottare le differenti tecniche di rappresentazione testuale e classificaizone, si è deciso di applicare una rielaborazione comune del testo nel seguente modo:
-rimozione di spazi bianchi
-conversione dei caratteri in lower case
-rimozione di stopwords
-rimozione della punteggiatura
-rimozione di URLS
-Tokenization
In fine, si creano due differenti attributi all'interno del dataset:
- "Text lemmatization": testo a cui viene applicata la lemmatizzazione alle parole
-"Text stemmed": testo a cui viene applicata la stemmatizzazione alle parole

Prima di iniziare la nostra analisi, abbiamo deciso di visualizzare attraverso Word Cloud, le parole più frequenti all'interno delle recensioni.



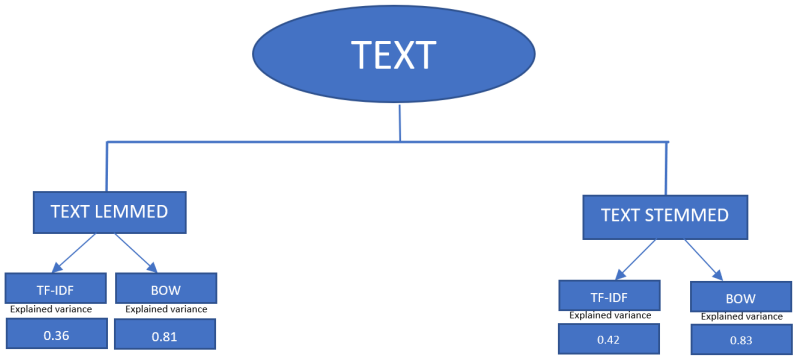
5 Text representation

Una volta rielaborato il testo, per poter applicare algoritmi di classifcazione supervisionata, si procede ad adottare due tecniche di rappresentazione testuale:
- Bag-of-Words: viene conteggiata la frequenza di ogni parola all'interno del testo ottenendo la term-document count matrix, ogni documento è rappresentato come count vector
- Tf-idf: tecnica che combina le euristiche di tf (conteggio delle volte in cui il termine t occorre nel documento d) e idf (logaritmo del rapporto tra N ed il numero di documenti d in cui appare il termine t)

Al termine dell'operazione si ottengono quindi due diverse tipologie di testi che a loro volta vengono rappresentati con due diverse tecniche.

5.1 Dimensionality reduction - SVD

Nonstante la riduzione iniziale, la mole di dati rimane consistente, rendendo difficoltosa dal punto di vista computazionale l'applicazione delle diverse analisi preposte.
In seguito ad una serie di tentativi, si è deciso di ridurre la dimensionalità ad un numero di 400 variabili, tramite l'utilizzo di Singular Value Decomposition (SVD). Essa, riduce le dimensioni della matrice di co-occorenza X tramite metodi dell'algebra lineare, in particolare ogni riga della matrice risultante U è la rappresentazione densa k-dimensionale di ogni parola nel vocabolario.
L'approccio utilizzato permette quindi una riduzione del peso computazionale mantenendo allo stesso tempo una buona interpretabilità dei testi, osservabile tramite varianza spiegata, per entrambe le tecniche di rappresentazione testuale utilizzate.



6 Supervised learning and classification

Il dataset viene suddiviso in training set e test set, con una proporzione rispettivamente del 70% e 30%, e vengono applicati i seguenti algoritmi:
- Adaboost
- Decision tree classiffier
- Gaussian Naive Bayes classifier
- Knn
- Random forest
- Logistic regression

Al fine di osservare le prestazioni dei modelli utilizzati, si sono prese in considerazione le metriche di accuracy, avg precision (media delle precison calcolate per ogni classe), avg recall (media delle recall calcolate per ogni classe) ed avg f-1 score(media delle f-1 score calcolate per ogni classe).

6.1 Adaboost

Il modello combina differenti classificatori (detti weak learners) insieme per aumentare l'efficacia della classificazione. Ogni classificatore viene addestrato utilizzando un set di campioni per l'addestramento ed il modello applica in modo iterativo i weak learners calcolandone un peso per ciascuno, il quale rappresenta la robustezza del classificatore.
L'algoritmo utilizzato viene impostato con un albero decisionale come weak learner.
In seguito si mostrano le performance del modello applicato per la classificazione delle recensioni in 3 classi (positive, negative e neutre) ed in 5 classi (ognuna per score) per ogni diversa combinazione di elaborazione e rappresentazione testuale.

3 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.51	0.53
Precision	0.51	0.53
Recall	0.51	0.53
F-1 measure	0.51	0.53

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.51	0.54
Precision	0.50	0.54
Recall	0.51	0.54
F-1 measure	0.50	0.54

5 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.37	0.38
Precision	0.36	0.37
Recall	0.37	0.38
F-1 measure	0.36	0.37

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.37	0.40
Precision	0.36	0.38
Recall	0.37	0.40
F-1 measure	0.35	0.39

6.2 Logistic regression

L'utilizzo di questo modello, anche se inzialmente pensato per tipi di classificazioni binarie, può essere applicato a casi di multi class classification con schema one-vs-rest, suddividendo il problema inziale in problemi di classificazione binaria multipla ed adottando una regressione logistica standard ad ogni sotto-problema.
In seguito si mostrano le performance del modello applicato per la classificazione delle recensioni in 3 classi (positive, negative e neutre) ed in 5 classi (ognuna per score) per ogni diversa combinazione di elaborazione e rappresentazione testuale.

3 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.61	0.63
Precision	0.61	0.63
Recall	0.61	0.63
F-1 measure	0.60	0.63

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.60	0.63
Precision	0.59	0.63
Recall	0.60	0.63
F-1 measure	0.59	0.63

5 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.47	0.49
Precision	0.46	0.48
Recall	0.47	0.49
F-1 measure	0.46	0.49

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.47	0.49
Precision	0.46	0.48
Recall	0.47	0.49
F-1 measure	0.46	0.48

6.3 Key nearest neighbors

Il terzo modello utilizzato per la classificazione delle recensioni è stato il KNN (key nearest neighbors). L'algoritmo non parametrico applica una classificazione basandosi sulla somiglianza delle caratteristiche, più un'istanza è vicina a un data point, più il knn li considererà simili ed assegnerà tale istanza ad una determinata classe.
L'algoritmo viene inizializzato con un numero di neighbours pari a 3 e poi pari a 5 per le diverse classificazioni effettuate.
Si riportano in seguito i risultati.

3 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.55	0.53
Precision	0.55	0.55
Recall	0.55	0.53
F-1 measure	0.55	0.53

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.54	0.53
Precision	0.55	0.54
Recall	0.54	0.53
F-1 measure	0.54	0.52

5 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.40	0.39
Precision	0.40	0.39
Recall	0.40	0.39
F-1 measure	0.39	0.38

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.39	0.38
Precision	0.40	0.39
Recall	0.39	0.38
F-1 measure	0.39	0.37

6.4 Gaussian Naive Bayes

Il classificatore gaussiano naive Bayes calcola la probabilità che un determinato elemento x appartenga ad una certa calsse c sulla base della funzione:

$P(x|\mathrm{c}) = P(x|\mathrm{c})P(c)/sumP(x|\mathrm{c})P(c)$

Si riportano in seguito i risultati ottenuti nelle diverse combinazioni.

3 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.44	0.50
Precision	0.46	0.51
Recall	0.44	0.50
F-1 measure	0.42	0.50

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.43	0.49
Precision	0.45	0.50
Recall	0.43	0.49
F-1 measure	0.41	0.49

5 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.26	0.36
Precision	0.30	0.36
Recall	0.26	0.36
F-1 measure	0.20	0.35

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.26	0.35
Precision	0.29	0.35
Recall	0.26	0.35
F-1 measure	0.21	0.33

6.5 Decision tree classifier

L'utilizzo del decision tree classifier, in un caso di classificazione, utilizza una struttura ad albero dove i nodi foglia rappresentano le classificazioni e le ramificazioni l'insieme delle proprietà che portano a quelle classificazioni.
Il modello viene inizializzato con parametro "Gini" per l'assegnazione della classe per ogni istanza ad ogni iterazione. Si riportano in seguito i risultati ottenuti.

3 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.68	0.70
Precision	0.68	0.70
Recall	0.68	0.70
F-1 measure	0.68	0.69

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.68	0.70
Precision	0.68	0.70
Recall	0.68	0.70
F-1 measure	0.68	0.70

5 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.56	0.57
Precision	0.55	0.57
Recall	0.56	0.57
F-1 measure	0.55	0.57

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.56	0.58
Precision	0.55	0.57
Recall	0.56	0.58
F-1 measure	0.55	0.57

6.6 Random forest classifier

Il Random forest classifier è costituito da un numero di alberi decisionali individuali che operano come un insieme. Ogni singolo albero nella foresta casuale

emette una previsione di classe e la classe con il maggior numero di voti diventa la previsione del nostro modello.
Il modello utilizzato viene inzializzato con un numero di alberi decisionali pari a 100, come di default, e utilizzato "Gini" come criterio di assegnazione della classe.
Si riportano in seguito i risultati ottenuti.

3 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.78	0.78
Precision	0.78	0.79
Recall	0.78	0.78
F-1 measure	0.78	0.79

Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.77	0.79
Precision	0.77	0.79
Recall	0.77	0.79
F-1 measure	0.77	0.79

5 CLASS CLASSIFICATION

Text stemmed		
Metrics	BoW	Tf-idf
Accuracy	0.63	0.66
Precision	0.64	0.66
Recall	0.63	0.65
F-1 measure	0.63	0.65

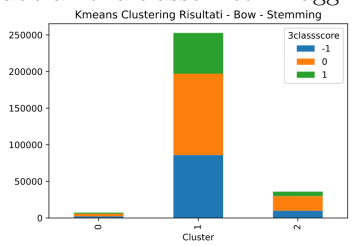
Text lemmed		
Metrics	BoW	Tf-idf
Accuracy	0.63	0.65
Precision	0.63	0.66
Recall	0.63	0.65
F-1 measure	0.63	0.65

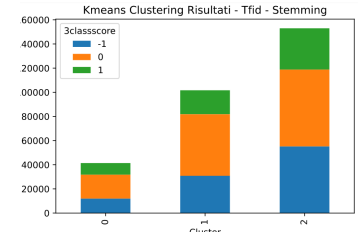
7 Clustering

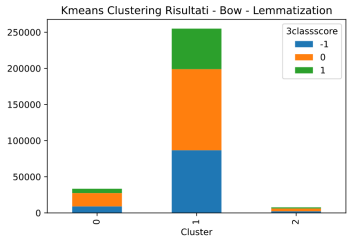
Come secondo apporccio per la classificazione delle recensioni si è deciso di utilizzare il text clustering.
A tal fine, abbiamo applicato l'algoritmo k-means il cui obiettivo è quello di minimizzare la distanza quadratica media euclidea degli oggetti dai loro centri di cluster.

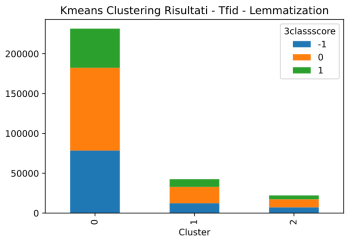
7.1 Clustering 3 classi

Nella classificazione in 3 classi, notiamo come nel testo rappresentato tramite BoW, in entrambi i casi (lemmatization e stemmattization), la classe con maggior frequenza è la 1. Al contrario per il TD-IDF, per il testo stemmatizzato abbiamo la classe 2 con maggior frequenza, mentre la 0 per quello lemmatizzato

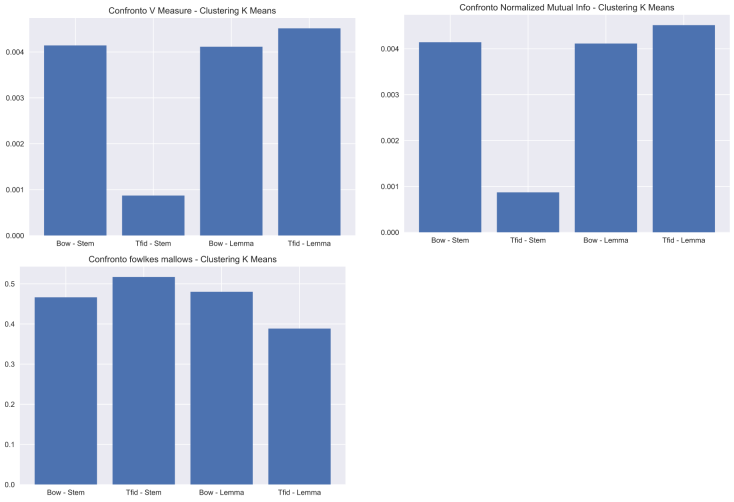








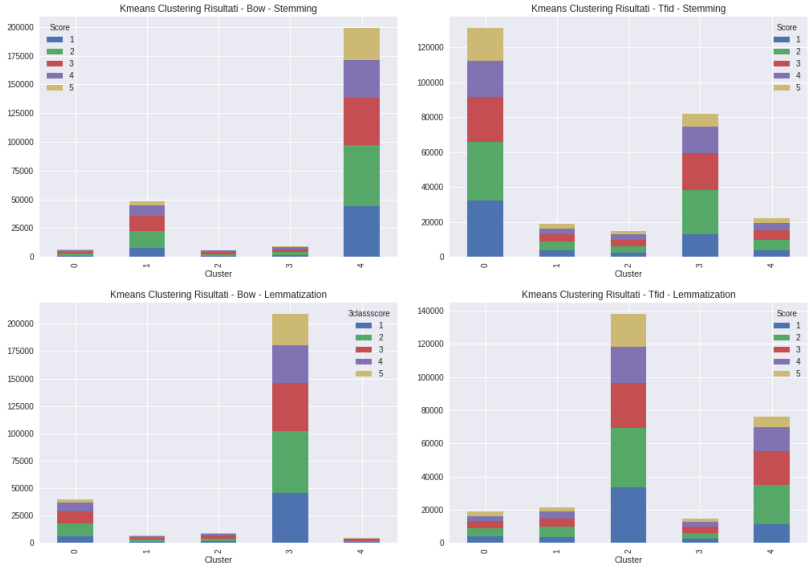
Confronti 3 classi



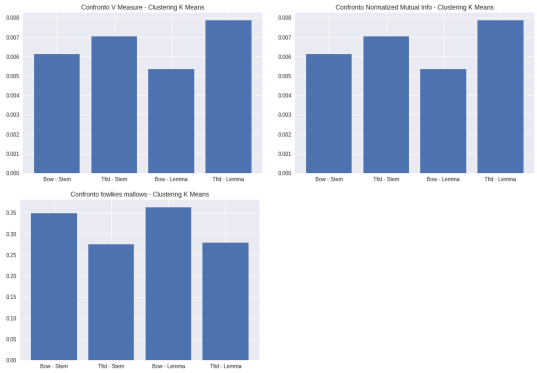
7.2 Clustering 5 classi

Per quanto riguarda questo tipo di clustering, si può facilmente notare come, nel primo caso (BoW stemming) le classi con maggior frequenza sono la numero 1 e la 4, mentre nel TD-IDF la classe 0 e 3.
Invece per quanto riguarda il testo lemmatizzato, per il BOW le classe con maggior frequenza sono la 0 la 3, mentre per il TD-IDF la classe 2 e 4.

Confronti 5 classi



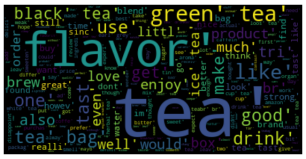
Confronti 5 classi



7.3 Word Cloud

Una volta effettuato il Text Clustering, abbiamo utilizzato nuovamente il metodo Word Cloud per visualizzare le parole più frequenti sia per le 3 classi che per le 5.
Ad esempio, per il clustering in tre classi,quanto emerge dal bag of word emerge dalle seguenti immagini.

Invece per il clustering in cinque classi,quanto emerge dal bag of word emerge dalle seguenti immagini.

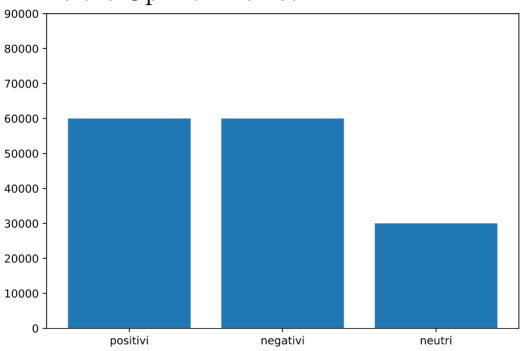


Lo stesso procedimento è stato applicato per gli altri algoritmi per i clustering a classi (si vedano i codici python)

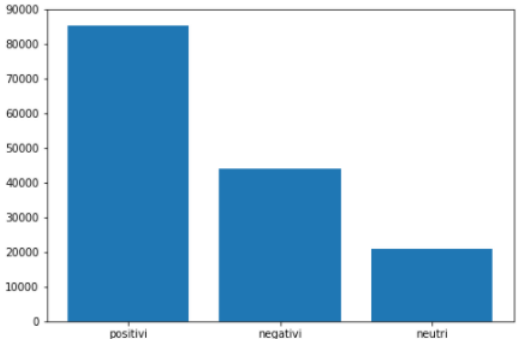
8 Sentiment analysis

Come metodo di classificazione delle recensioni, in base al contenuto del testo, si utilizza la tecnica di sentiment analysis.
Prima di procedere all'applicazione di algoritmi, il testo viene sottoposto ad una modellazione che prevede l'eliminazione di URLs,HTML tags, spazi bianchi, conversione in lower case e lemmatizzazione.
Per il calcolo della sentiment analysis viene utilizzato l'algoritmo "Opinion Lexicon", che assegna ad ogni recensione un punteggio che, se maggiore al valore zero,si considera la recensione come positiva, se minore di zero negativa e neutra se uguale a zero. La funzione di assegnazione del punteggio viene scritta in modo tale da invertire la polarità della frase nel caso in cui si dovesse trovare una negazione all'interno di essa.
Si osserva inizialmente il numero di recensioni positive, negative e neutre rilevate dall'algoritmo e messo a confronto con i valori iniziali ottenuti dopo il bilanciamento del dataset. Graficamente si osserva come l'algoritmo sia propenso ad una classificazione positiva delle recensioni a discapito delle recensioni negative e neutre.

Before Opinion Lexicon



After Opinion Lexicon



Si osserva in seguito la capacità dell'algoritmo di classificare il testo in 5 classi, ognuna per score. Per fare questo si sono osservati i punteggi ottenuti ed eliminati diversi outliers che distorcevano il range iniziale di valori portando quindi lo score nel range [-13;+22]. Tramite una funzione di rescale si sono riscalati i valori nel range [1;5] e arrotondati per poter poi eseguire un confronto con la feature Score.
Si è poi creata una matrice di confusione per poter ossevare l'accuratezza del modello:

9 Conclusioni

Osservando i risultati ottenuti, tramite le diverse combinazioni di elaborazione, rappresentazione tesutale e classificiazione possiamo dare dei commenti sulle tecniche utilizztae.
Il metodo della classificazione supervisionata, in particolare nella calssificazione a 3 calssi, riporta accuratezze più elevate. Si osserva inoltre che la stemmatizzazione applicata alle parole nel testo insieme ad una rappresentazione Bow,

permettono il raggiungimento, anche se minimo, di valori di accuratezza più alta.
Si evidenzia che il modello migliore sia "Random Forest", con una accuratezza del 78.0% (3 class classification, Stemmed Bow) e 63% (5 class classification, Stemmed Bow) .
Osservando i risultati ottenuti dalla classificazione non supervisionata, si può affermare che l'algorimto k-means non classifica correttamente le recensioni. Difatti, dopo il bilanciamento del dataset ed osservgando i grafici riportati, l'algoritmo classifica i dati in classi slbilanciate in tutte le combinazioni di elaborazione e rappresentazione del testo.
L'utilizzo della sentiment analysis, analogamente all'apporccio non supervisionato, non classifica in modo ottimale i dati presenti nel dataset. In particolare l'algoritmo utilizzato tende ad assegnare un valore positivo alle recensioni e nel caso di classificazione in 5 classi, nonostante la gestione degli outliers e il ridimensionamento dei valori, assegna questi ultimi in modo differente rispetto allo Score ufficiale.

10 Bibliografia

https://towardsdatascience.com/understanding-random-forest-58381e0602d2
https://www.kdnuggets.com/2020/01/decision-tree-algorithm-explained.html
https://machinelearningmastery.com/multinomial-logistic-regression-with-python/
https://www.developersmaggioli.it/blog/classificatori-non-lineari-classificazione-k-nn/