Università degli studi di
Milano - Bicocca

Text Mining & Search

Final Project

Amazon Reviews Classification

Autori:
Alex Ceccotti - 790497 - a.ceccotti@campus.unimib.it
Stefano Fiorini - 778379 - s.fiorini2@campus.unimib.it
Davide Pecchia - 793290 - d.pecchia1@campus.unimib.it

January 15, 2019

Abstract

Il sistema di reviews di Amazon è uno dei punti di forza del colosso dell'e-commerce mondiale. Permette infatti la collaborazione da parte degli utenti, i quali sono sempre in grado di fornire un feedback dopo aver effettuato un acquisto di qualsiasi genere. Le recensioni sono libere, ogni persona può esprimere un suo giudizio senza vincoli, a parte il mantenimento di un linguaggio adeguato. É interessante dunque capire, a partire semplicemente dal testo, quale sia la tipologia del prodotto revisionato (ambito tech, video games, giardinaggio...). Per fare ciò si è partiti da una grande quantità di recensioni pubblicate su Amazon, più di un milione, divise in 11 categorie differenti. Dopo una prima fase di pulizia dei dati e di preprocessign del testo, il dataset è stato splittato in training, test e validation set, al fine di poter procedere alla fase di implementazione del modello, in cui è stata addestrata una rete neurale relativamente semplice, composta da 4 hidden layers. A seguire è stata eseguita una fase di fine tuning dei parametri al fine di selezionare quelli ottimali per il modello. Una volta raggiunto un buon risultato, l'algoritmo è stato validato sul test set; sono state infine definite delle metriche di valutazione in modo da poter valutare la qualità del modello e proporre possibili miglioramenti da apportare.

1 Introduzione

Amazon è il più grande sito di e-commerce al mondo. Ha rivoluzionato radicalmente le abitudini di acquisto dei consumatori, creando un nuovo paradigma: per effettuare un acquisto non è più necessario recarsi fisicamente in uno store, bensì si può svolgere questa operazioni comodamente da casa. Esistono moltissime tipologie di prodotti acquistabili, dai video games ad oggetti per il giardinaggio, da articoli di abbiagliamento a prodotti tech. La strategia di Amazon si basa anche su un sistema di "collaborazione" da parte degli utenti: dopo ogni acquisto infatti, è possibile lasciare una review in base all'esperienza appena condotta, in modo che gli altri utenti possano avere una quantità di informazioni maggiore, al fine di effettuare un acquisto consapevole. In base al testo delle reviews dunque è possibile risalire alla tipologia di articolo che si sta valutando e classificare di conseguenza l'ambito di appartenzenza (tech, beauty, video games...).

1.1 Obiettivo

Obiettivo del progetto è quello di classificare correttamente le reviews scritte su Amazon dagli utenti, assegnando a ciascuna la corretta categoria di appartenenza. Il fine ultimo è quello di verificare per ogni prodotto se la classificazione delle reviews ad esso associate coincide con la categoria di appartenenza del prodotto stesso.
É stato dunque scelto, per il progetto, un task di *classification*.

2 Dataset

Per le analisi è stata utilizzata una raccolta di reviews riguardanti prodotti che si possono trovare su Amazon; queste sono reperibili al link `http://jmcauley.ucsd.edu/data/amazon/`. Sono state utilizzate 11 categorie, principalmente per motivi computazionali. Le categorie selezionate sono:

- Automotive (20.473 reviews)
- Beauty (198.502 reviews)
- Cell phones & accessories (194.439 reviews)
- Digital music (64.706 reviews)
- Grocery and Gourmet food (151.254)
- Office products (53.258 reviews)
- Patio, lawn & garden (13.272 reviews)
- Pet supplies (157.836 reviews)
- Tools & home improvement (134.476 reviews)
- Toys & games (167.597 reviews)
- Video games (231.780 reviews)

In totale le reviews utilizzate per le analisi sono 1.387.593, tutte in lingua inglese. Tutti i dati a disposizione sono in formato .json.

2.1 Data Manipulation

Essendo i dati in formato .json è stato necessario definire una funzione che permettesse di ottenere un dataframe pandas, in modo da permettere una migliore gestione del dato. Sono stati quindi importati tutti gli 11 dataset e successiavemente sono stati uniti in un unico dataframe. Le variabili disponibili per ogni review sono: *reviewerID*, *asin*, *reviewerName*, *helpful*, *reviewText*, *overall*, *summary*, *unixReviewTime*, *reviewTime*, *y*. Ai fini dell'analisi, sono state tenute in considerazione solo la varaibile ***reviewText***, contenente il testo della recensione, e la variabile ***y***, indicante la categoria di appartenenza (vairabile di tipo numerico con valori appartenenti all'intervallo 0-10, essendo 11 il numero totale di categorie considerate). Una volta estratte le 2 colonne, il subset è stato salvato in una nuova variabile, in modo da avere il dataset pronto da utilizzare per le analisi successive.

3 Approccio Metodologico

A partire dal nuovo subset ottenuto nella fase di data manipulation, è stata inizialmente svolta una prima fase di preprocessing del testo, in modo da poterlo snellire e renderlo più interpretabile nella successiva fase, quella di classification, in cui viene svolto effettivamente il task di classificazione delle reviews in base alla categoria di appartenenza.

3.1 Preprocessing del testo

Inizialmente sono state svolte alcune operazioni di pulizia, ovvero è stato reso minuscolo tutto il testo, è stata rimossa la punteggiatura e sono state eliminate le stopwords tramite la libreria **nltk** - natural language toolkit - che mette a disposizione un dizionario di stopwords inglesi.
Nella fase successiva, è stato effettuato un processo di lemmatization, tecnica utilizzata per ridurre ad un lemma, ovvero la forma canonica di un set di parole, tutti i termini presenti nel testo. In particolare, la lemmatization è utile per eliminare le declinazioni delle varie parole, che sono presenti all'interno del linguaggio naturale (ad esempio, ridurre al lemma dei vocaboli permette di eliminare le differenze tra singolare e plurale), tenendo anche conto nel contesto in cui si trovano i termini. Questa tecnica è stata preferita a quella di stemming in quanto, dopo aver effettuato diversi tentativi, i risultati raggiunti sono stati simili, ma non altrattanto ottimali. Computazional-

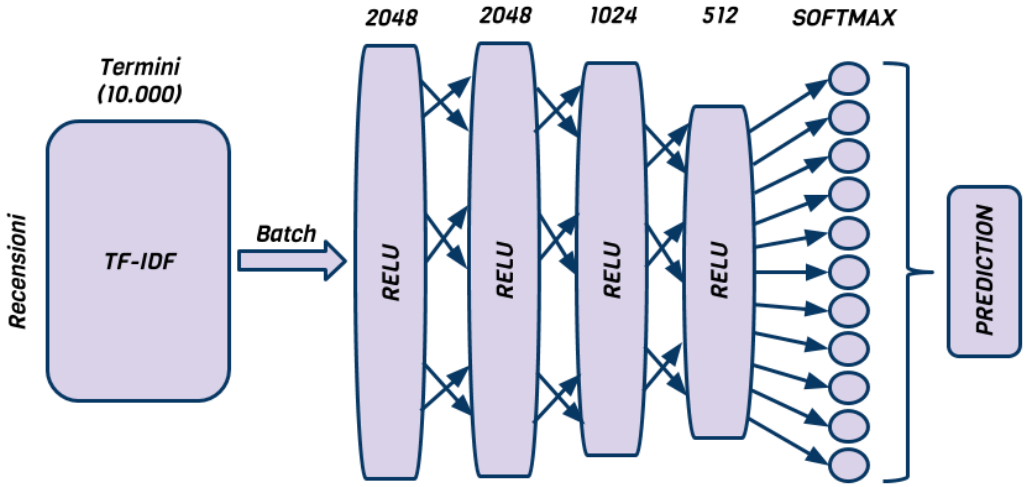
mente parlando inoltre, si è notato che gli algoritmi di stemming richiedevano molto più tempo rispetto a quelli di lemmatization, dunque i primi sono stati selezionati per l'analisi. Una volta ottenuto l'output del processo, è stata costruita una matrice con numero di righe pari al numero di osservazioni e numero di colonne pari a 10000 (numero di features considerate in modo da ridurre la sparsità della matrice, attraverso la funzione TfidfVectorizer della libreria sklearn). Nella matrice così ottenuta, ogni cella assume valore pari alla tf-idf del termine (colonna) nella recensione (riga).
Per concludere la fase di preprocessing è stato fatto un dump della matrice ottenuta e delle labels tramite la libreria joblib. É stata eseguita questa procedura a causa dell'alto sforzo computazionale richiesto in questa fase di preprocessing; il dump è quindi utile per evitare di dover ripetere nuovamente tutti gli step una volta trovata la configurazione ottimale.

3.2 Classification

Inizialmente, in questa fase, il dataset è stato suddiviso in 2 parti:

- training set + validation set (90% delle osservazioni)
- test set (10% delle osservazioni)

Le percentuali di composizione dei differenti set sono state decise in base al numero di osservazioni disponibili (10% del dataset corrisponde a 138.760 reviews).
Una volta splittato il dataset, per la classificazione viene utilizzata una rete neurale fully connected con 4 hidden layers (composti rispettivamente da 2048 - 2048 - 1024 - 512 neuroni). Come funzione di attivazione per i neuroni degli hidden layers viene utilizzata la "relu", mentre lo strato di output ha 11 neuroni a cui viene applicata la funzioni di attivazione softmax. L'output della NN sarà dunque la probabilità di appartenenza di ogni review ad una determinata classe. Per quanto riguarda l'ottimizzazione, è stato utilizzato l'ottimizzatore 'adam' mentre come loss function la 'categorical crossentropy'. Per la valutazione viene utilizzata l'accuracy come metrica.
Una volta definita la rete, si procede con la successiva fase di addestramento del modello. Le epoche utilizzate sono 3 ed il batch size definito è 1024. Quest'ultimo è relativamente grande, ma è stato definito in questo modo per 2 principali motivazioni:



- il numero di osservazioni relativamente alto consente comunque di ottenere ottime prestazioni dal momento che i parametri vengono aggiornati molteplici volte per ogni epoca.
- le tempistiche di computazione si riducono notevolmente.

É stata utilizzata inoltre parte del set di osservazioni (20%) come validation, sul quale verranno valutate accuracy e loss function alla fine di ogni epoca. Una volta ottenuti e raccolti i risultati, si è passati ad una successiva fase di valutazione attraverso l'uso di opportune metriche.

4 Risultati e Valutazioni

I risultati ottenuti sono stati, in termini di accuracy:

- 0.9817 sul training set
- 0.9322 sul validation set
- 0.9314 sul test set

Si può notare come si siano ottenuti dei buoni risultati su tutti i vari set di analisi, nonostante ci sia un leggero overfitting sul training set.
Per quanto riguarda altre metriche di valutazione, i risultati ottenuti sono riassunti nella tabella sottostante:

	precision	recall	f1-score	support
0	0.97	0.96	0.97	15100
1	0.79	0.66	0.72	2068
2	0.94	0.94	0.94	19965
3	0.93	0.93	0.93	19216
4	0.98	0.98	0.98	6535
5	0.87	0.92	0.90	5318
6	0.78	0.78	0.78	1263
7	0.93	0.94	0.93	15850
8	0.88	0.87	0.88	13575
9	0.92	0.93	0.92	16759
10	0.96	0.96	0.96	23111
micro avg	0.93	0.93	0.93	138760
macro avg	0.90	0.90	0.90	138760
weighted avg	0.93	0.93	0.93	138760

Nel complesso dunque le varie metriche risultano elevate, sia in termini di precision che di recall. Infatti anche la metrica F1-score risulta essere complessivamente molto positiva (0.93).

5 Conclusioni

Ottenute e valutate tutte le metriche, è possibile concludere che il modello riesce a eseguire con successo il task richiesto. Ci si aspetta dunque che, analizzata una qualsiasi review, essa venga classificata correttamente all'interno dell'ambito di appartenenza con un'elevata probabilità. É possibile migliorare ulteriormente la qualità del modello in primis utilizzando un maggior numero di reviews per il training, ma ciò necessita uno sforzo computazionale maggiore e richiede di conseguenza una maggiore capacità in questi termini. Sarebbe anche possibile implementare una rete neurale (o comunque un classificatore) più articolato, ma ciò non è stato ritenuto necessario dati gli ottimi risultati raggiunti.