

Analisi di Yelp Open Dataset

Lorenzo Vainigli

Corso di Intelligenza Artificiale a.a. 2019/20

Laurea Magistrale in Informatica

Università di Bologna

Indice

1	Introduzione	2
2	Dati	2
3	Obiettivi	3
4	Strumenti	3
5	Classificazione delle recensioni	3
5.1	Creazione del modello	3
5.2	Addestramento e validazione	4
5.3	Miglioramento delle performance	5
5.4	Risultati finali	5
6	Raggruppamento degli utenti	7
6.1	Distribuzione di <i>average_stars</i>	8
6.2	Distribuzione di <i>fans</i>	8
6.3	Distribuzione di <i>review_count</i>	9
7	Raggruppamento dei locali	9
7.1	Migliori e peggiori	9
7.2	Categorie	10
7.3	Ubicazione	11

1 Introduzione

In questo documento sono riportati i risultati dell'analisi di *Yelp Open Dataset* [1], una base di dati che contiene informazioni su esercizi commerciali di varie categorie e presenti in diverse città degli Stati Uniti e del Canada. Il dataset è stato esaminato al fine di costruire un modello predittivo basato su una rete neurale, per produrre statistiche aggregate e trovare i valori migliori e peggiori per alcuni tipi di dato.

I file di questo progetto sono disponibili nel repository dell'autore su GitHub [2].

2 Dati

I dati di *Yelp Open Dataset* sono utilizzabili per uso personale, educativo o accademico, sono disponibili in formato *JSON* e sono divisi in alcuni file:

- *business.json* (153 MB): contiene le informazioni relative agli esercizi commerciali tra cui ubicazione e categoria.
- *review.json* (6,33 GB): contiene i testi delle recensioni includendo l'identificativo dell'utente che ha scritto la recensione e l'esercizio commerciale oggetto della recensione.
- *user.json* (3,27 GB): contiene i dati associati ai singoli utenti, inclusi gli identificativi degli amici.

Il database contiene anche i file *tip.json*, *checkin.json* e *photos.json*, ma non sono stati presi in considerazione per lo sviluppo di questo progetto.

Caricamento dei dati Per motivi di performance non è stato possibile analizzare tutto il contenuto dei file *review.json* e *user.json*, poiché troppo grandi, mentre è stato possibile esaminare *business.json* interamente.

3 Obiettivi

Lo scopo del progetto prevede l'analisi dei dati per studiare la loro struttura e il loro contenuto, al fine di estrapolare osservazioni interessanti su di essi. Non si tratta solo di aggregare record o trovare valori minimi, massimi o medi, ma di applicare anche tecniche di elaborazione del linguaggio naturale e machine learning. In particolare, le finalità del progetto richiedono:

- T1) il riconoscimento automatico di una review positiva o negativa (sezione 5, codice in `notebooks/reviews_classification.ipynb`);
- T2) il raggruppamento degli utenti in base alle loro preferenze o comportamento sulla piattaforma (sezione 6, codice in `notebooks/users.ipynb`);
- T3) il raggruppamento automatico dei locali in base a criteri di similitudine data una certa località (sezione 7, codice in `notebooks/businesses.ipynb`).

La cartella `notebooks` contiene altri file in cui vengono effettuate ulteriori analisi del contenuto del dataset.

4 Strumenti

Per conseguire gli obiettivi sopra citati i dati sono stati elaborati in *Python* con ampio utilizzo delle librerie *pandas*, *numpy*, *matplotlib* e *tensorflow* sulla piattaforma *Jupyter Notebook*.

5 Classificazione delle recensioni

Grazie al dataset *review.json* è stato possibile costruire un modello predittivo basato su una rete neurale.

Il file `review_model_test.ipynb` contiene uno script per utilizzare la rete neurale già addestrata, in cui l'utente può testare frasi a sua scelta.

5.1 Creazione del modello

I valori di input sono stati calcolati applicando il *word embedding* al testo delle recensioni. I valori di output per ogni input sono stati calcolati in base al numero di stelle associato alla recensione e trasformato in un vettore

binario per permettere al modello di effettuare una classificazione basata su categorie (es. "4 stelle" è stato codificato con il vettore $[0, 0, 0, 1, 0]$).

Dopo numerosi tentativi, provando diverse configurazioni di livelli e parametri diversi, è stata scelta la seguente configurazione:

```
1 embedding_dim = 128
2 max_length = 256
3
4 model = tf.keras.Sequential([
5     tf.keras.layers.Embedding(vocab_size, embedding_dim,
6                               input_length=max_length),
7     tf.keras.layers.GlobalAveragePooling1D(),
8     tf.keras.layers.Dropout(0.4),
9     tf.keras.layers.Dense(5, activation='softmax')
10 ])
```

5.2 Addestramento e validazione

Sono state selezionate le prime 10.000 recensioni, senza alcun criterio di filtraggio, effettuando una divisione 80%-20% per costruire, rispettivamente, training set e test set. In fase di addestramento è stato aggiunto un *early stopping*.

I valori di *accuracy* risultanti sono 79,8% per il training set e 63,7% per il test set (Figura 1). Sarebbe stato molto interessante selezionare 100.000 recensioni anziché solo un decimo di queste, ma non è stato fatto per limiti di capacità computazionali dell'elaboratore.

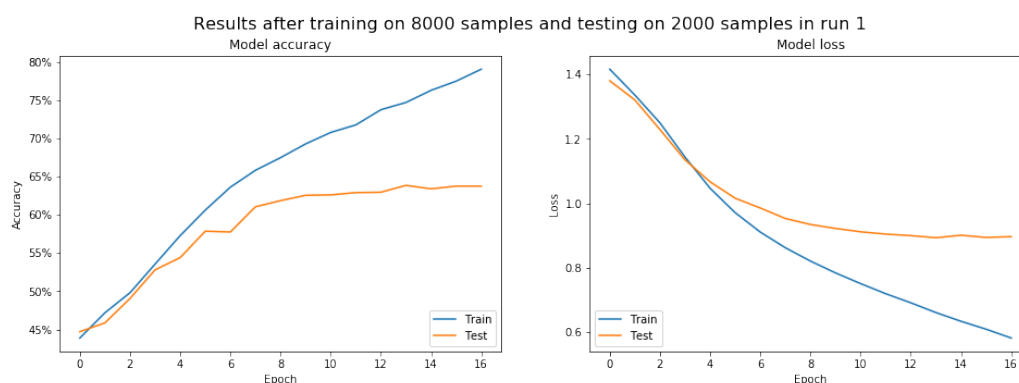


Figura 1: Valori di *accuracy* e *loss* per il primo run, ovvero prima di aver effettuato il filtraggio sui dati.

5.3 Miglioramento delle performance

Considerando che cambiando i parametri o la composizione del modello i risultati non miglioravano in modo significativo, si è scelto di effettuare un filtraggio sui dati eliminando quelli che producevano valori di *loss* troppo alti. L'assunzione dietro a questa scelta sta nel fatto che questi valori di input-output potrebbero non essere molto veritieri.

Per ogni istanza dei 10.000 record scelti si è calcolato il valore di *loss* e si è deciso di scartare quelle che presentavano un valore più alto di 2.

I valori di *accuracy* risultanti dopo questa operazione sono 88,9% per il training set e 70,3% per il test set (Figura 2).

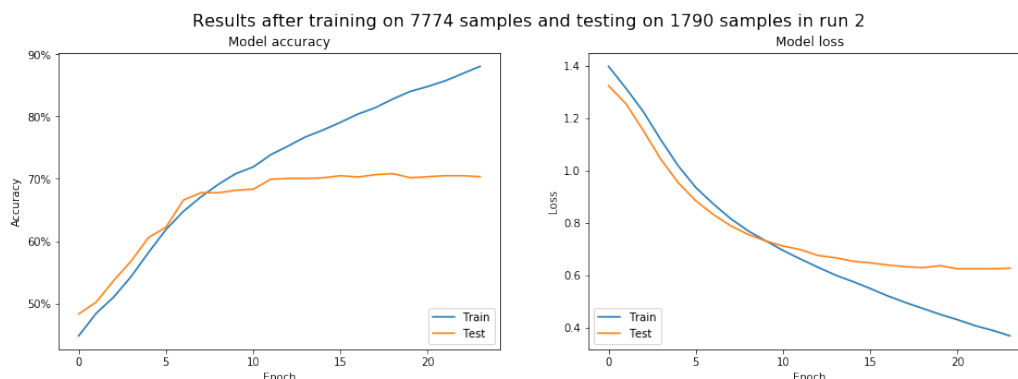


Figura 2: Valori di *accuracy* e *loss* per il secondo run, ovvero prima di aver effettuato il filtraggio sui dati dove sono stati eliminati gli esempi con valori di *loss* alta.

5.4 Risultati finali

Di seguito sono riportati dei grafici che mostrano i dati sui quali si può effettuare una valutazione finale sul modello per il riconoscimento automatico di una recensione.

Al fine di una classificazione tra positive e negative possiamo valutare come positive le recensioni a cui vengono attribuite 4 o 5 stelle e come negative quelle a cui ne vengono attribuite 1 o 2 stelle. Il caso delle 3 stelle è arbitrario. Osservando i grafici mostrati nelle figure 3, 4, 5 e 6 possiamo verificare come l'accuratezza del modello sia migliorata.

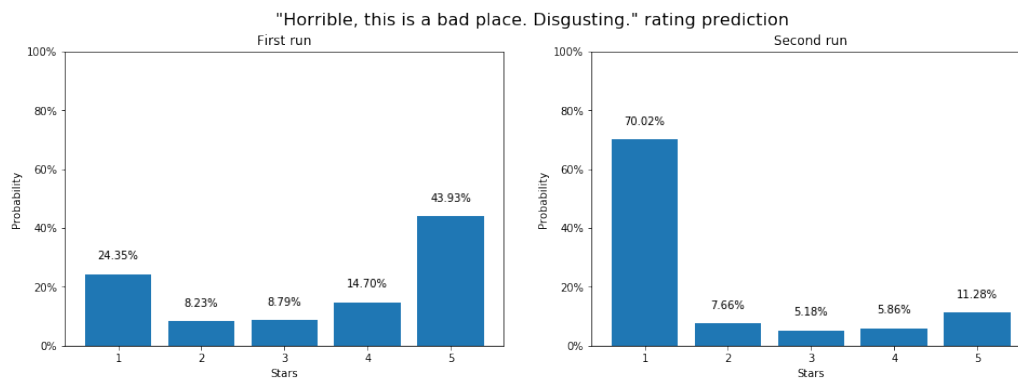


Figura 3: Questa frase è sicuramente una delle peggiori che si possa scrivere a proposito di un locale. Al primo run il modello assegnava erroneamente 5 stelle ma con una notevole incertezza, mentre nel secondo run la scelta migliore (1 stella) è nettamente superiore alle altre.

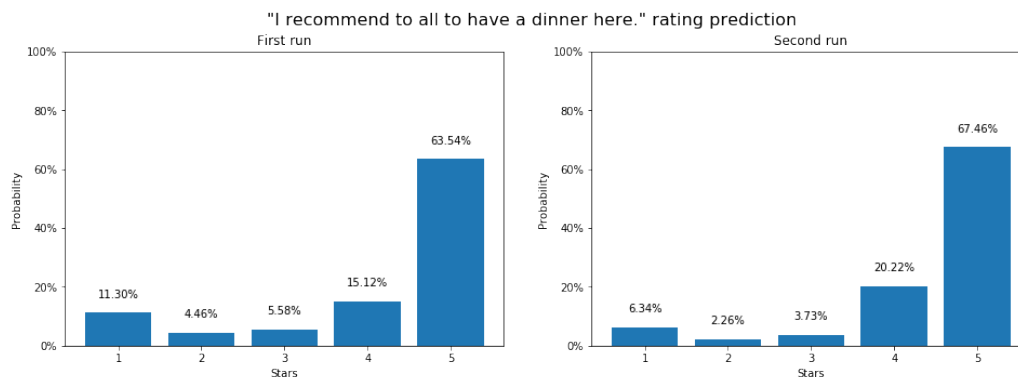


Figura 4: Questa è senza dubbio una frase positiva e anche al primo run si ha una scelta netta per l'opzione 5 stelle, che comunque viene rafforzata ulteriormente nel secondo run. Sarebbe anche stato lecito aspettarsi 4 stelle.

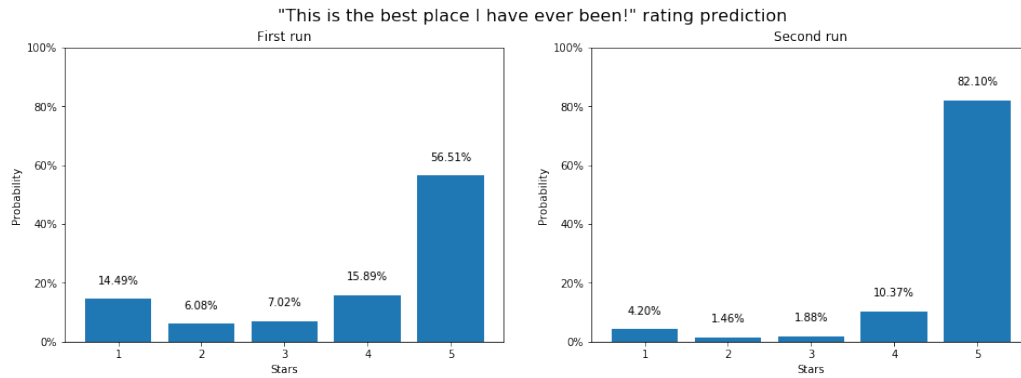


Figura 5: Recensione assolutamente positiva. Già con il primo run si ha una scelta chiara, ma con il secondo l'82% di probabilità è un risultato estremamente corretto.

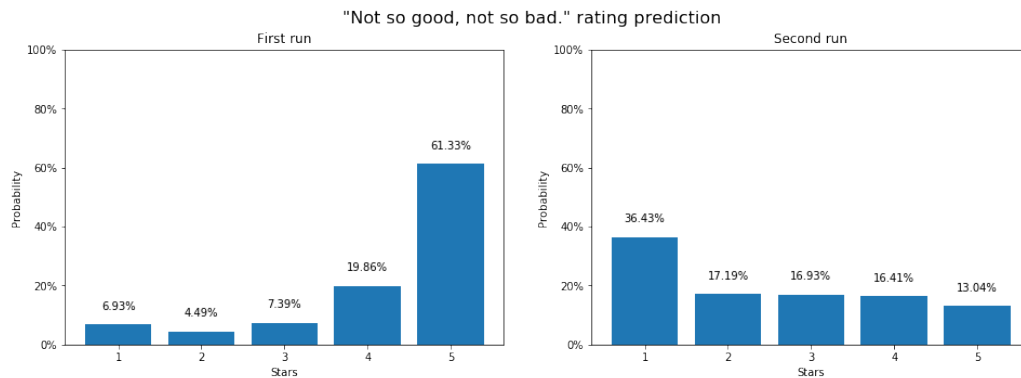


Figura 6: Questa frase è una delle più interessanti poiché rappresenta un equilibrio per cui sarebbero preferibili 3 stelle (poiché si tratta della scelta media). Purtroppo in questo caso il modello non è molto accurato neanche dopo il secondo run, infatti la scelta di attribuire una stella è sicuramente sbagliata.

6 Raggruppamento degli utenti

Per questioni di performance e di limiti di memoria per l'elaborazione sono stati caricati solo i primi 100.000 record del file *user.json*, che è composto dai campi *average_stars*, *fans*, *friends*, *name*, *review_count*, *useful*, *user_id*, e

altri campi di minore importanza.

Per gli utenti è stato ritenuto utile esaminare la distribuzione dei valori per i campi *average_stars*, *fans* e *review_count*,

6.1 Distribuzione di *average_stars*

Questo campo rappresenta la media delle stelle assegnate alle recensioni del singolo utente e, a differenza di tutti gli altri campi, presenta una distribuzione simile a una gaussiana.

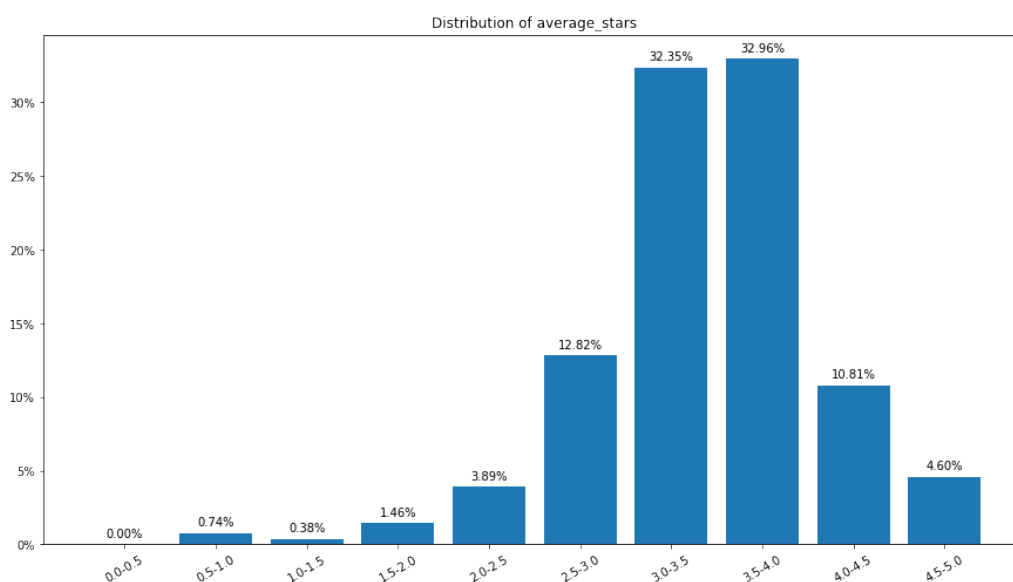


Figura 7: Distribuzione dei valori del campo *average_stars* in termini percentuali.

6.2 Distribuzione di *fans*

Il 98,96% degli utenti presenta un numero di fan tra 0 e 109 e l'81,71% di questi ha meno di 6 fan. Questi valori portano alla conclusione che la user base di questo dataset è prevalentemente composta da utenti che interagiscono con una piccola cerchia di altri utenti.

6.3 Distribuzione di *review_count*

Questo valore dà una precisa indicazione del contributo che un utente apporta al dataset. Dall'analisi emerge che il 99,25% degli utenti ha scritto meno di 1032 recensioni, ma è una percentuale plausibile. Molto più interessante è esaminare la segmentazione degli utenti per quanto riguarda coloro che hanno scritto meno di 100 recensioni.

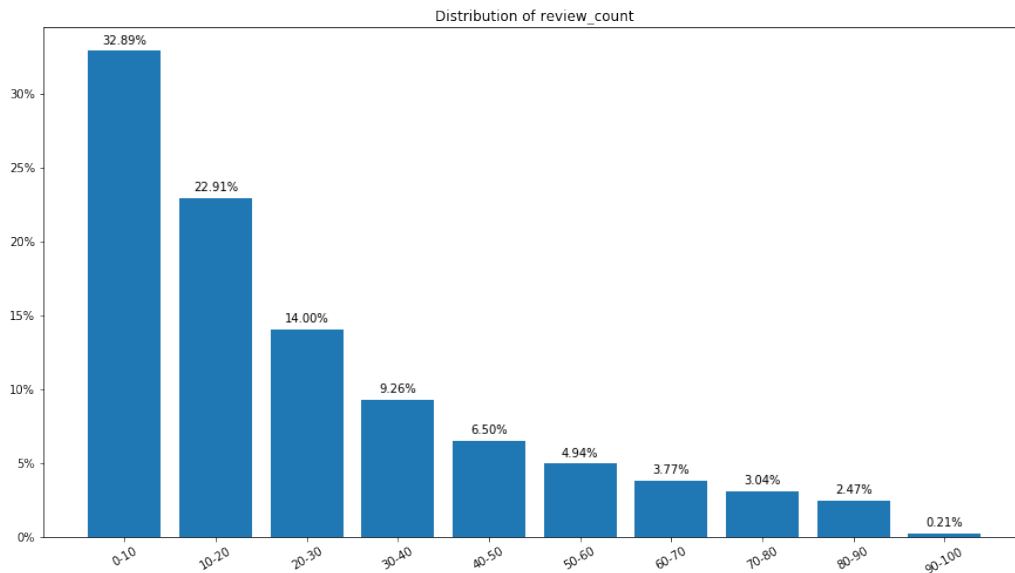


Figura 8: Distribuzione dei valori del campo *review_count* per valori tra 0 e 100 in termini percentuali.

7 Raggruppamento dei locali

Il file *business.json* contiene 209.393 record, ognuno composto da 14 campi: *address*, *attributes*, *business_id*, *categories*, *city*, *hours*, *is_open*, *latitude*, *longitude*, *name*, *postal_code*, *review_count*, *stars* e *state*.

7.1 Migliori e peggiori

Per questa classificazione sono state prese in considerazione il numero di stelle assegnate a ogni esercizio commerciale e il numero di recensioni ricevuto. Si presume che, a parità di stelle, più il numero di recensioni è alto, più

questo valore sia affidabile.

Sono state analizzate quattro delle categorie più diffuse: *Restaurants*, *Shopping*, *Health & Medical* and *Automotive*.

Little Miss BBQ (Phoenix, AZ), **Brew Tea Bar** (Las Vegas, NV) e **Cocina Madrigal** (Phoenix, AZ) sono i migliori ristoranti secondo la media delle stelle e il numero di recensioni ricevute, mentre **McDonald's** (Las Vegas, NV), **KFC** (Avondale, AZ) e **McDonald's** (Fort Mill, SC) sono i peggiori. Tra i negozi catalogati come *Shopping*, i migliori sono **Eco-Tint** (Las Vegas, NV), **Studio 21 Tattoo Gallery** (Las Vegas, NV) e **FINO for MEN** (Las Vegas, NV). I peggiori sono **DIRECTV** (Phoenix, AZ), **Bank of America Store and Heritage Center** (Charlotte, NC) e **Teleflora Fresh Flowers** (Las Vegas, NV).

Bangkok Thai Spa Massage (Las Vegas, NV), **Simply Skin Las Vegas** (Las Vegas, NV) e **Richards Cosmetic Surgery, Med Spa & Laser Center** (Las Vegas, NV) sono i luoghi migliori per la categoria *Health & Medical*. Sempre per quanto riguarda questa categoria, i luoghi peggiori sono **SilverScript Medicare** (Phoenix, AZ), **Apria Healthcare** (Henderson, NV) e **OptumCare Primary Care - Deer Valley** (Phoenix, AZ).

I migliori esercizi commerciali per *Automotive* sono **Eco-Tint** (Las Vegas, NV), **Precision Window Tint** (Henderson, NV) e **DC Auto Luxury Window Tinting** (Las Vegas, NV). I peggiori sono **Phoenix Car Rental** (Phoenix, AZ), **LendingTree** (Charlotte, NC) e **Seller Networks** (Las Vegas, NV).

Considerando le città, **Las Vegas** è quella dove si possono trovare gli esercizi commerciali migliori, considerando queste quattro categorie, seguita da **Phoenix**.

7.2 Categorie

Le categorie presenti sono in totale 1.207 e le più diffuse sono **Restaurants** (13,5%), **Shopping** (10,7%), **Home Services** (8,2%), **Food** (7,7%), **Health & Medical** (6,8%), **Beauty & Spas** (13,5%), **Local Services** (5,5%), **Automotive** (4,6%), **Nightlife** (4,4%) e **Event Planning & Services** (13,5%).

7.3 Ubicazione

Le città in cui si trovano i locali sono 1.306. La maggior parte dei locali si trova a **Las Vegas** (15%), seguita da **Toronto** (10%), **Phoenix** (10%), **Charlotte** (5%) e **Scottsdale** (4%).

Se si effettua il raggruppamento per Stato, allora il 29% si trova in **Arizona** (AZ), il 19% in **Nevada** (NV), il 17% in **Ontario** (ON), l'8% in **Ohio** (OH) e l'8% in **North Carolina** (NC). I restanti sono divisi tra altri Stati.

8 Conclusioni

Yelp Open Dataset è sicuramente un valido dataset se si vuole sviluppare progetti di data analysis oppure se si ha bisogno di testare la propria applicazione, ma diventa ancora più interessante in quanto può essere utilizzato, in particolare la parte che raccoglie le recensioni degli utenti, per costruire modelli di elaborazione del linguaggio naturale e di sentiment analysis.

Riferimenti

- [1] *Yelp Open Dataset*. URL: <https://www.yelp.com/dataset/>.
- [2] *Analysis of Yelp Open Dataset*. URL: <https://github.com/lorenzovngl/analysis-of-yelp-open-dataset>.