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

I file di questo progetto sono disponibili nel repository dell'autore su GitHub [1].

2 Dati

Yelp Open Dataset [2] è una base di dati che raccoglie informazioni su esercizi commerciali di varie categorie. I dati 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.
- *tip.json* (263 MB): contiene dei suggerimenti che gli utenti scrivono a proposito degli esercizi commerciali. Possono essere visti come delle brevissime recensioni.

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

3 Obiettivi

Lo scopo del progetto prevede l'analisi dei dati al fine di 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 NLP e machine learning. In partcolare, le finalità del progetto richiedono:

T1) il riconoscimento automatico di una review positiva o negativa;

T2) il raggruppamento degli utenti in base alle loro preferenze o comportamento sulla piattaforma;

T3) il raggruppamento automatico dei locali in base a criteri di similitudine data una certa località.

A queste, ne sono state aggiunte altre:

T4) analisi dei singoli file JSON;

T5) classificazione dei locali migliori e peggiori per ogni categoria;

T6) locali aperti nelle vicinanze dell'utente;

T7) utenti con le recensioni più affidabili (comparando il loro voto alla media dei voti di un determinato locale);

T8) migliori recensioni e suggerimenti (tips) per un locale.

4 Strumenti

Per conseguire gli obiettivi sopra citati i dati sono stati elaborati in Python con l'utilizzo di Jupyter Notebook [3].

5 Sviluppo

Per ogni obiettivo (o target) T* è stato creato un notebook presente nella cartella `notebooks`:

T1) `reviews_classification.ipynb`;

T2) `users_grouping.ipynb`;

T3) `businesses_grouping.ipynb`;

T4) `business.ipynb`, `review.ipynb`, `tip.ipynb`, `user.ipynb`;

T5) `best_and_worst_businesses.ipynb`;

T6) `closest_opened_businesses.ipynb`;

T7) `best_reviewers.ipynb`;

T8) `best_business_tips.ipynb`.

5.1 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.

6 Risultati

6.1 Esercizi commerciali

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*.

6.1.1 Migliori e peggiori

Per questa classificazione sono stete prese in considerazione il numero di stelle assegnate a ogni esercizio commerciale e il numero di recensioni ricevuto. Si presume che, a partità 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**.

6.1.2 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%).

6.1.3 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.

6.2 Utenti

Per questioni di performance e di limiti di memoria per l'elaborazione sono stati caricati sono 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*,

6.2.1 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.

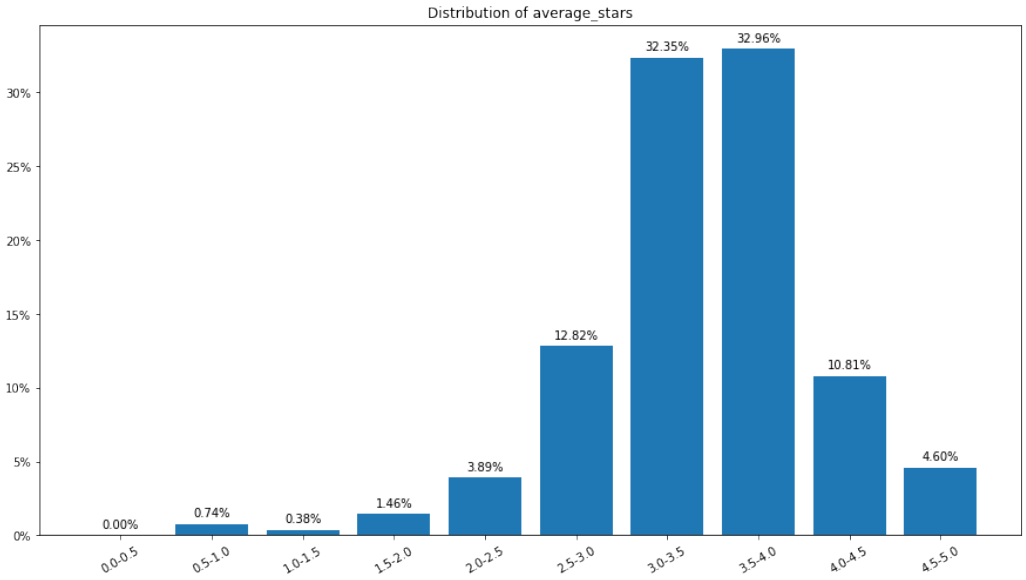

Figura 1: Distribuzione dei valori del campo *average_stars* in termini percentuali.

6.2.2 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.

6.2.3 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.

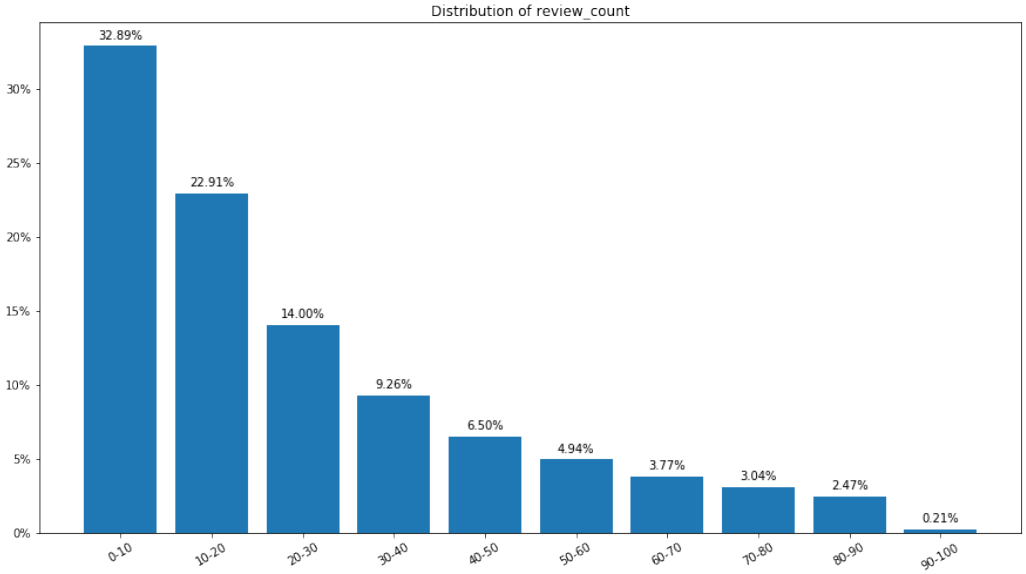

Figura 2: Distribuzione dei valori del campo *review_count* per valori tra 0 e 100 in termini percentuali.

6.3 Classificazione delle recensioni

6.3.1 Configurazioni della rete neurale

```
model = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size, embedding_dim,
    input_length=max_length),
    tf.keras.layers.GlobalAveragePooling1D(),
    tf.keras.layers.Dense(24, activation='relu'),
    tf.keras.layers.Dense(1, activation='sigmoid')
])
model.compile(loss='binary_crossentropy', optimizer='adam',
    metrics=['accuracy'])
```

```
model = tf.keras.Sequential([
    tf.keras.layers.Embedding(vocab_size, embedding_dim,
    input_length=max_length),
    tf.keras.layers.GlobalAveragePooling1D(),
    tf.keras.layers.Dense(24, activation='relu'),
    tf.keras.layers.Dense(12, activation='sigmoid'),
    tf.keras.layers.Dense(6, activation='sigmoid'),
    tf.keras.layers.Dense(3, activation='sigmoid'),
```

```
    tf.keras.layers.Dense(1, activation='sigmoid')
])
model.compile(loss='binary_crossentropy', optimizer='adam',
    metrics=['accuracy'])
```

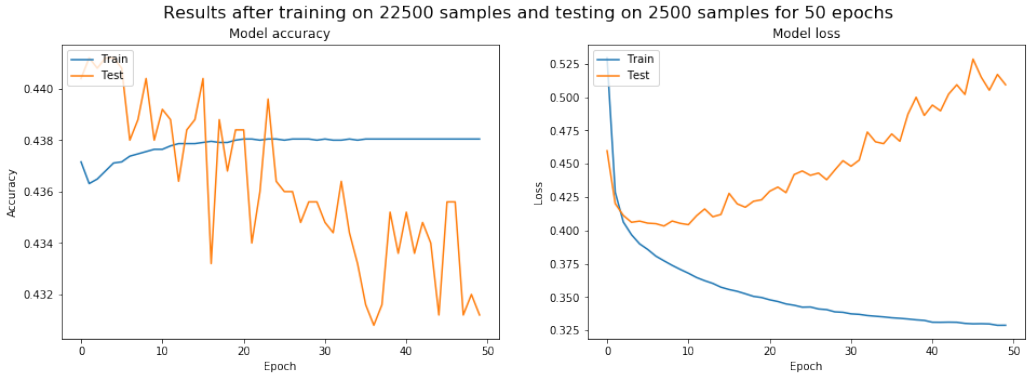

Figura 3: Risultati di *accuracy* e *loss* dopo un training della rete neurale in configurazione 1 con 22.500 esempi e testing con 2.500 esempi per 50 iterazioni.

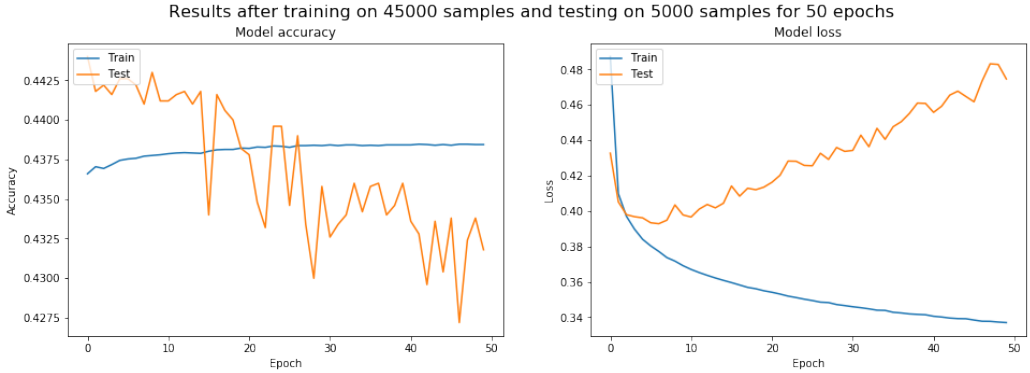

Figura 4: Risultati di *accuracy* e *loss* dopo un training della rete neurale in configurazione 1 con 45.000 esempi e testing con 5.000 esempi per 50 iterazioni.

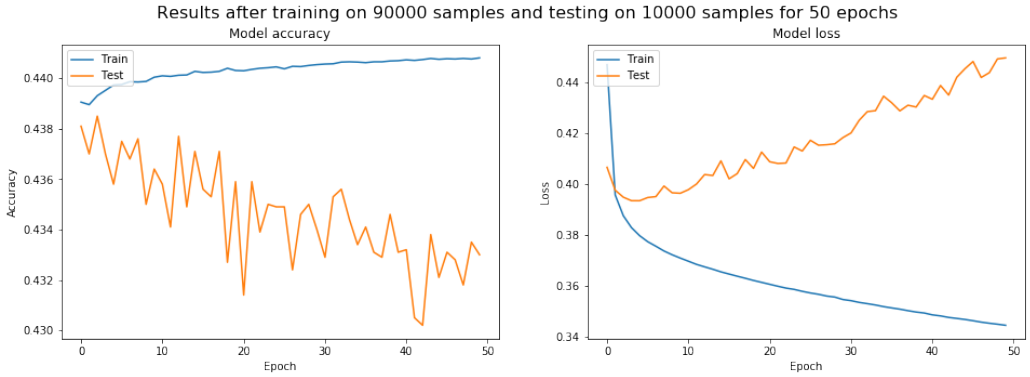

Figura 5: Risultati di *accuracy* e *loss* dopo un training della rete neurale in configurazione 1 con 90.000 esempi e testing con 10.000 esempi per 50 iterazioni.

7 Conclusioni

Riferimenti

[1] *Analysis of Yelp Open Dataset*. URL: `https://github.com/lorenzovngl/ai-project`.

[2] *Yelp Open Dataset*. URL: `https://www.yelp.com/dataset/`.

[3] *Jupyter Notebook*. URL: `https://jupyter.org/`.