
Politecnico di Bari

Dipartimento di Ingegneria Elettrica e dell'Informazione

Corso di Laurea Magistrale in Computer Science Engineering (Information Systems)

---

Final Project in

Artificial Intelligence and Machine Learning

# Raccomandazioni musicali per completamento di playlist

Docenti:

Prof.: Tommaso di Noia
Prof: Vito Walter Anelli
Ing: Claudio Pomo
Ing: Antonio Ferrara

Studente:

Giuseppe Spadavecchia

A.A 2020/2021

## *Abstract*

In questo lavoro sviluppa un sistema di raccomandazione per suggerimenti musicali utilizzando come piattaforma di riferimento Spotify. Tra le tecniche con cui sviluppare un sistema di raccomandazione si è deciso di usare il collaborative filtering.

## *Introduzione*

La presenza di piattaforme, in molti casi web, in cui sono presenti una grande quantità di informazioni, ha reso necessario la creazione di sistemi di raccomandazione per il suggerimento di contenuti.
I sistemi di raccomandazione possono essere realizzati attraverso quattro diversi approcci:

- Collaborative Filtering
- Content Based
- Knowledge Based
- Hybrid Recommender System

I sistemi di raccomandazioni collaborative filtering possono essere ideati basandosi su modelli o su memoria, per lo più usando l'algoritmo K-Nearest Neighbor (KNN) [1]. L'approccio generale in questo caso è quello di trovare una correlazione tra le valutazioni di più utenti.
Nell'approccio Content Based si considerano gli attributi descrittivi degli item per effettuare delle raccomandazioni.
I sistemi di raccomandazione Knowledge-Based sono spesso usati in ambiti in cui gli item da raccomandare risultano essere poco valutati, infatti sono spesso usati per raccomandare beni di lusso, automobili, beni finanziari e immobiliari. Il processo di raccomandazione prevede di calcolare la somiglianza tra i requisiti dell'utente e le descrizioni degli item, facendo uso delle basi di conoscenza.
Infine, un'ulteriore tipologia di RS è quella ibrida che combina un sistema di raccomandazione Content Based e Collaborative Filtering. Un esempio di sistema di questo tipo è quello usato da Netflix [2].

## *Lavori correlati*

Spotify è una delle più famose piattaforme di streaming musicale on-demand, che presenta una selezione di brani appartenenti a diverse case ed etichette discografiche e di podcast realizzati dagli utenti.
Per gestire la grande quantità di dati presenti e per rendere più semplice la loro fruizione da parte degli utenti, viene usato un sistema di raccomandazione.
Spotify nel suo sistema di raccomandazione fa uso di tre modelli: il collaborative filtering, l'elaborazione del linguaggio naturale (NLP) e la modellazione di dati audio.

Il primo modello viene usato per generare la playlist ‘Discover Weekly' considerando utente tra di loro simili, attraverso la generazione di profili relativi a diversi utenti.
L’utilizzo del NLP permette di individuare termini specifici presenti nei brani e cercare di predire il significato delle frasi. Questo permette di fornire raccomandazioni più efficaci.
La modellazione di dati audio fa utilizzo di una CNN e il suo compito principale è quello di considerare i dati relativi ad artisti e canzoni permettendo anche a quelli che normalmente non rilevanti di esserlo [3][4].
Questo lavoro non considera la raccomandazione di un’intera playlist come effettivamente avviene in Spotify, come nei casi delle ‘Daily Mix’ o della ‘Discover Weekly’, ma data una canzone in ingresso di raccomandarne altre.

## *Dataset e feature*

Il dataset utilizzato per effettuare le analisi è stato ottenuto presso [5], come file .csv. I dati presentano diverse feature ottenibili tramite l’estrazione dei dati da Spotify stesso. Tra le feature più rilevanti si possono annoverare quelle legate alle caratteristiche audio, come ‘loudness’ e ‘tempo’, e quelle legate al brano come ‘track_id’, ‘track_name’, ‘artists’ e ‘genre’. Particolare importanza è data a ‘track_id’, un identificativo univoco usato da Spotify assegnato ad ogni singolo brano.
Affinché si potesse modellare con semplicità i dati si è deciso di memorizzarli in una struttura dati adeguata, tra quelle possibili si è scelto il dataframe pandas.
L’analisi del dataset ha messo in rilevanza come esistessero diversi brani con genere differente, ma stesso identificativo. Si è deciso di trattare questi elementi come dei duplicati e, pertanto, sono stati eliminati nella fase di preprocessing.
Un ulteriore manipolazione del dataset è risultata necessaria per generare il modello per effettuare le raccomandazioni. Si è creato un dataset copia dell’originale in cui venivano eliminate tutte le feature non numeriche, come il nome del brano, dell’artista e il genere, e si effettuata una normalizzazione z-score la cui formula è la seguente $x = \frac{x-\mu}{\sigma}$, in cui $\mu$ è la media e σ la deviazione standard, per le feature di ‘loudness’ e ‘tempo’. Si è deciso di utilizzare la feature ‘id’ come indice eliminandola successivamente da questa copia.
Questa elaborazione iniziale ha permesso la creazione di un modello che considera unicamente le caratteristiche musicali di ciascun brano.

## *Metodi*

Per effettuare le raccomandazioni si è deciso usare la tecnica del collaborative filtering. Nei dati presenti, come qualsiasi altro tipo di dato ricavabile direttamente da Spotify, non compare alcuna feature relativa al rating, in quanto la piattaforma stessa non lo prevede in una maniera simile come accade per le piattaforme di streaming di film o serie TV. Per ovviare a questo tipo di problematica si è deciso di utilizzare un collaborative filtering con

un approccio basato su memoria che consiste nel trovare gli elementi più vicini ad uno dato in input. Per raggiungere questo scopo si è usato l'algoritmo K Nearest Neighbors (KNN).
L'algoritmo KNN prevede di effettuare una classificazione o regressione dato un punto, trovando quelli a lui più vicini. Nel considerare i punti più vicini si possono fare uso di diverse metriche per calcolare la distanza come quella di Manhattan: $d(x,y) = \sqrt{\sum_{i=1}^{m} (x_i - y_i)^2}$ o quella euclidea $d(x,y) = \sum_{i=1}^{m} |x_i - y_i|$. Nel caso di utilizzo dell'algoritmo KNN per effettuare una classificazione questo indica a quale delle diverse classi appartiene il dato fornito in input, mentre nel caso della regressione si restituisce la media delle distanze dei KNN come prediction [6].
Nel caso di questo lavoro si è utilizzato il metodo fornito da 'sklearn' noto come 'Nearest Neighbors' [7]. Quest' ultimo prevede il settaggio di diversi parametri come l'algoritmo da usare per effettuare la ricerca e le metriche da utilizzare per calcolare la distanza. Presenta inoltre diversi metodi tra i quali quelli usati sono stati il metodo 'fit' e 'kneighbors'. Il metodo 'fit' prevede di adattare lo stimatore dei nearest neighbors al dataset fornito in input sotto forma di matrice sparsa. Il metodo 'kneighbors' trova i k, il cui valore è fornito come input, vicini al punto fornito in ingresso.
Questo permette di trovare i k elementi più vicini a uno dato che permette di indicare, a differenza di un normale caso di classificazione o regressione, quanto un elemento suggerito sia simile a quello fornito.
Nell'applicazione dell'algoritmo KNN si è deciso di usare i seguenti settaggi, l'algoritmo di ricerca usato è quello 'brute force' e la metrica usata è quella di 'Minkowski' la quale nel caso di due feature ricade nel caso della distanza euclidea e per la scelta del numero di vicini si è scelto di considerarne solo 10.
Dopo la creazione del modello con i parametri suddetti si effettua l'adattamento del modello i dati che vengono rimodellati in una matrice sparsa compressa.
Le raccomandazioni vengono effettuate fornendo in ingresso un particolare brano di cui si fornisce unicamente l'id utilizzato come indice per la ricerca. Fornendo questo dato, insieme al numero di raccomandazioni e quindi di vicini voluti si ottengono gli indici di questi ultimi e le distanze tra questi elementi e il dato fornito in input. Con l'utilizzo degli indici si formano le raccomandazioni di cui si elimina il primo elemento che risulta essere sempre quello fornito in input e sono ordinate in maniera crescente di distanza, pertanto il primo elemento è quello più vicino e, quindi, il più simile.
Per effettuare le raccomandazioni si è deciso di considerare tre sotto insiemi del dataset originale in base a tre generi: 'movie', 'jazz' e 'anime'. Il brano utilizzato come input per le raccomandazioni è scelto in maniera casuale in tutti i casi.
Generate le raccomandazioni queste vengono stampate a video considerando il nome del brano e dell'artista corrispondente.

```
La traccia musicale considerata è Jet Rocket eseguita da LiSA
                                                  name           artists
33049                                       BREAK IT!     Mamoru Miyano
30465                                       STARTRAiN          Amatsuki
33544              Live For Life - Ookamitachi no Yoru -             Aimi
29199                                  Idola No Circus            IKASAN
116816            Quitate la Ropa (Remix) [feat. Juanka]  Sammy & Falsetto
30056   Gimme! Revolution (From "Gonna Be the Twin-Tai...      Maaya Uchida
43204                            Netoge Haijin Sprechchor          kradness
27516                                    シルエット feat.MICO          Kobasolo
27101                                         Wake up!              AAA
```

*Figura 1 - Raccomandazioni per utente 'anime'*

```
La traccia musicale considerata è Couleur eseguita da Chantal Goya
                                                name               artists
86147                                Amo Soltanto Te        Andrea Bocelli
60305                                       Hey Adam       Mandolin Orange
148784                                Cradle and All        Audra McDonald
160767  You're Not Finished Yet (feat. Maggie Reed)      The Belonging Co
59959                        House of the Risin' Sun             Bob Dylan
56714                                   Stone Woman  Charlotte Day Wilson
151256                                     Wanna Be          Robby Benson
43354                            Sora Mo Toberuhazu        Mika Nakashima
7112                          Christmas Makes Me Cry       Kacey Musgraves
```

*Figura 2 - Raccomandazioni per utente 'movie'*

```
La traccia musicale considerata è Point to Point eseguita da Animals As Leaders
                 name                  artists
103635       Breakbeat          Slightly Stoopid
175016             1993              Manila Killa
47472     Sentient Oona              Thee Oh Sees
176173         Fictions                   Shallou
39011    String Electro                      Moby
12579           Really?  Prince Daddy & the Hyena
9169        Spontaneous              Flying Lotus
35882   Love of Strings                      Moby
25452            Winter                      3LAU
```

*Figura 3 - Raccomandazioni per utente 'jazz'*

## *Esperimenti e Risultati*

Per valutare il modello si è deciso di generare due utenti fittizi per due dei generi considerati precedentemente nell'effettuare le raccomandazioni, 'movie' e 'anime'. Di questi due utenti si va a considerare sia quali siano i loro brani preferiti che eventuali artisti preferiti.
In quanto non sono presenti dati relativi ai rating e non è possibile usare metriche come il 'Mean Absolute Error' (MAE) e il 'Root Mean Square Error' (RMSE). Inoltre, queste metriche risultano essere inadeguate per un caso come quello di questo lavoro che considera le migliori K raccomandazioni. Pertanto per valutare la bontà del sistema si sono utilizzate le metriche di Precision e Recall.
La Precision viene  calcolata come il rapporto tra la somma del numero di elementi rilevanti e il numero di raccomandazioni, mentre il Recall come il rapporto tra la somma del numero di elementi rilevanti e il numero totale di elementi usati per effettuare il test [1].

Per considerare il numero di elementi rilevanti si verifica se per ogni elemento presente tra quelli di test e per ogni elemento raccomandato se ci sia una corrispondenza. Questa informazione è memorizzata come un 1 se la corrispondenza esiste, 0 altrimenti.
La Precision permette di sapere quanto le raccomandazioni siano in linea con i gusti dell'utente, mentre il Recall dà informazioni sul rapporto dei veri positivi indicando quanti siano degli elementi testati quelli rilevanti.
Queste metriche vengono calcolate sia per le canzoni preferite che per gli artisti preferiti dai due utenti creati. I risultati ottenuti vengono visualizzati sia sotto forma numerica, ma anche attraverso un grafico come mostrato in figura 4.

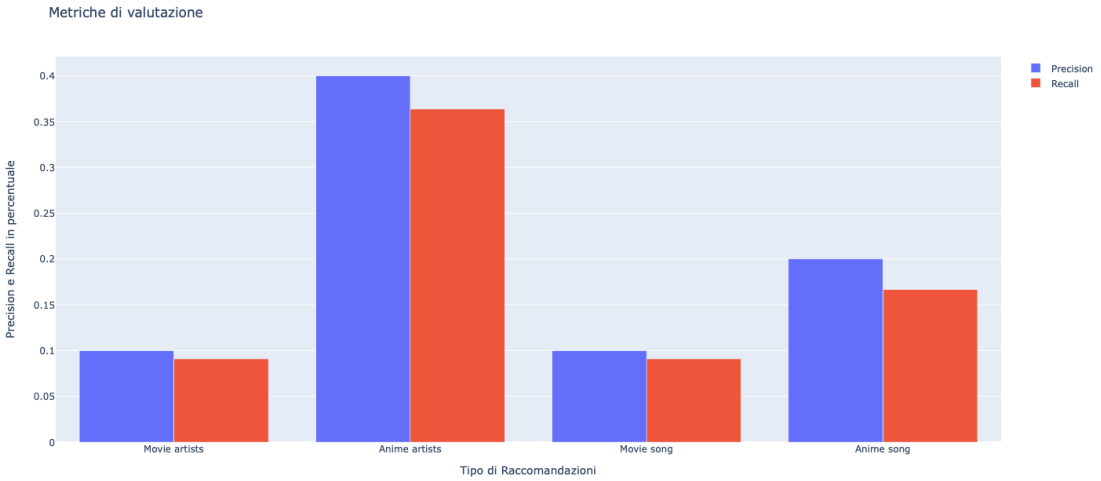


*Figura 4 - Grafico delle metriche di valutazione*

Come si può notare dal grafico i valori di Precision e Recall sono tra loro piuttosto simili, ma c'è una sostanziale differenza tra i due utenti. Questo è dovuto al metodo con cui le raccomandazioni vengono effettuate. Infatti, il modello tiene unicamente conto delle caratteristiche audio non andando a considerare il genere musicale del brano e l'artista in quanto il modello non può trattare dati in formato stringa. Quello che risulta è che le raccomandazioni non presentano unicamente brani appartenenti ad un unico genere o solo quelli eseguiti da un unico artista. Sebbene questo possa sembrare un demerito del modello va considerato come nelle metriche di un sistema di raccomandazione siano importanti la novità di ciò che viene raccomandato, per riuscire a suggerire elementi nella long tail e non solo i più popolari e la diversità, che indica che non bisogna suggerire elementi presenti unicamente in un unico sottoinsieme. Queste valutazioni sono considerate unicamente legate al genere in quanto per le caratteristiche audio risultano, invece, essere simili al brano fornito in input.

## ***Conclusioni***

Il sistema si dimostra abile nel suggerire i K possibili brani successivi ad uno fornito, permettendo il continuamento dell'ascolto. Come evidenziato dai risultati vengono raccomandati anche elementi non facenti riferimento allo stesso genere del brano fornito in input riuscendo a fornire valori di Precision e Recall accettabili. Come sviluppi futuri si può pensare di estendere il modello cercando di considerare anche dati come il genere per aumentare la precisione delle raccomandazioni,però abbandonando l'algoritmo KNN e sempre considerare  la vicinanza tra oggetti come elemento di somiglianza, ma usando come algoritmo uno di clustering come il K-Means.

## *Riferimenti*

[1]Materiale didattico del corso di Artificial Intelligence and Machine Learning dell'A.A. 2021/2022 tenuto da Tommaso Di Noia presso il Politecnico di Bari.
[2]Charu C. Aggraval. 2016. Reccomender Systems: The Textbook
[3]https://medium.com/datadriveninvestor/behind-spotify-recommendation-engine-a9b5a27a935
[4]https://towardsdatascience.com/how-spotify-recommends-your-new-favorite-artist-8c1850512af0
[5]https://www.kaggle.com/code/iqbalbasyar/spotify-genre-classification/data
[6]https://www.youtube.com/watch?v=0p0o5cmgLdE
[7]https://scikit-learn.org/stable/modules/generated/sklearn.neighbors.NearestNeighbors.html?highlight=nearest+neighbor#sklearn.neighbors.NearestNeighbors.kneighbors