MOvie recommender system

Zagaria Riccardo

Matricola: 760475

A.A. 2024 - 2025

E-mail: [r.zagaria18@studenti.uniba.it](mailto:r.zagaria18@studenti.uniba.it)

Link GitHub: https://github.com/venicebeachl/IconProject.git

Sommario

[1) Introduzione 2](#_Toc186732501)

[2) Creazione del Dataset 3](#_Toc186732502)

[3) Ricerca 4](#_Toc186732503)

[3.1) Algoritmo Random Walk 4](#_Toc186732504)

[3.2) Risultati di Precisione e Recall 4](#_Toc186732505)

[3.3) Possibili Miglioramenti 5](#_Toc186732506)

[4) Apprendimento Supervisionato 6](#_Toc186732507)

# Introduzione

L’obiettivo di questo progetto è quello di sviluppare un Recommender System in grado di consigliare dei film all’utente. L’origine delle raccomandazioni può essere Content-Based, ovvero basato sulla similarità tra due o più prodotti, film in questo caso, o User-Based, ovvero si basa sulla similarità tra due o più utenti.

## Requisiti funzionali

Il progetto è stato realizzato in Python, ad oggi uno dei linguaggi di programmazione più utilizzati, poiché questo linguaggio mette a disposizione numerose librerie che permettono di esaminare facilmente i dati, o di realizzare facilmente dei grafici, utili per analizzare statisticamente i risultati ottenuti da determinate soluzioni.

La versione di Python utilizzata è la versione 3.12.8. L’IDE utilizzato per lo sviluppo del progetto è, invece, Visual Studio Code. Si può, però, utilizzare un IDE qualsiasi.

Tra le librerie di Python utilizzate, troviamo invece:

* **matplotlib**, per la realizzazione dei grafici;
* **numpy**, per la gestione degli array;
* **pandas**, per l’importazione dei dataset nel formato *.csv*;
* **seaborn**, per la realizzazione di altri grafici;
* **sklearn**, per l’apprendimento automatico;
* **os**, per l’interazione con il sistema operativo;
* **cornac**, per lo sviluppo di recommender system.
* **Imblearn**, per l’oversampling tramite SMOTE;

# Creazione del Dataset

Il dataset utilizzato è il MovieLens 100K.

MovieLens 100K è stato preprocessato per unificare informazioni su utenti, film e valutazioni. Questo processo ha incluso i seguenti passaggi chiave:

* **Caricamento dati**: I dataset relativi agli utenti, ai film e alle valutazioni sono stati importati da file separati.
* **Unione dei dati**: Attraverso operazioni di merge, sono state integrate le informazioni demografiche degli utenti, i dettagli sui film (titoli e generi) e le valutazioni esplicite fornite dagli utenti.
* **Gestione dei dati mancanti**: Sono stati identificati e trattati i valori mancanti per garantire l'integrità e la qualità del dataset finale.
* **Trasformazione e pulizia**: I dati sono stati convertiti in un formato adatto per le analisi successive, eliminando le colonne non necessarie e standardizzando i tipi di dati.

I dati mancanti sono stati gestiti utilizzando tecniche come l'imputazione basata su medie o modalità per le colonne numeriche e categoriche, rispettivamente. In alcuni casi, i record con dati mancanti in campi critici sono stati rimossi per mantenere l'integrità del dataset. Successivamente, i dati sono stati trasformati in un formato adatto per l'analisi, standardizzando i valori e garantendo la coerenza tra le colonne.

## 2.2) Descrizione delle Feature

Durante il progetto, sono state selezionate feature rilevanti per ogni macroarea al fine di ottimizzare le prestazioni dei modelli:

* **Età e Genere**: Utilizzate nelle reti bayesiane per stimare la probabilità che un utente valuti positivamente un film, integrando informazioni demografiche.
* **Generi dei Film**: Usati per identificare le preferenze degli utenti sia nel Logic-Based Recommender che in Random Walk. Questi hanno permesso di individuare film con caratteristiche simili a quelli già apprezzati.
* **Similarità tra Utenti e Film**: Calcolata tramite metriche come Pearson o Cosine, queste sono state utilizzate nel KNN per raccomandare film basati sulle preferenze degli utenti più simili.
* **Valutazioni Esplicite**: Fondamentali in tutte le tecniche per addestrare e valutare i modelli, consentendo di identificare pattern di comportamento e feedback.

Queste feature sono state scelte per bilanciare la semplicità e l'efficacia, considerando i vincoli del dataset MovieLens 100K e gli obiettivi del progetto.

# Ricerca

L'obiettivo principale della macroarea della ricerca è stato esplorare tecniche basate sulla rappresentazione grafica per identificare film potenzialmente interessanti per gli utenti. Dopo una valutazione approfondita delle soluzioni possibili, è stato scelto l'approccio del **Random Walk**, in quanto in grado di sfruttare la similarità tra i film all'interno di un grafo.

Nel grafo, i nodi rappresentano i film, mentre le connessioni tra i nodi sono basate sulla similarità tra i film stessi, calcolata in base ai generi cinematografici. Per ottenere questa similarità, è stata impiegata la tecnica del **TF-IDF** (Term Frequency-Inverse Document Frequency) combinata con la **cosine similarity**, garantendo una rappresentazione accurata e dettagliata dei generi. Questa scelta ha permesso di catturare le affinità tra i film, anche quando non strettamente legati da una semplice somiglianza nel titolo o trama.

## 3.1) Algoritmo Random Walk

L'algoritmo di **Random Walk** inizia da un set di film che l'utente ha valutato positivamente, e successivamente esplora il grafo seguendo le connessioni tra i film simili. Per migliorare la coerenza delle raccomandazioni, è stato introdotto un **bias** che favorisce i film già apprezzati dall'utente. Questo bias consente di orientare la ricerca verso i gusti personali dell'utente, migliorando così la pertinenza delle raccomandazioni. Inoltre, per contrastare problemi di bassa precisione, è stato ridotto il numero di passi del **Random Walk**, limitando la ricerca ai film che sono apparsi più frequentemente durante il cammino.

Un altro passo importante è stato l'inclusione di un filtro che seleziona solo i film con alta frequenza e che sono fortemente simili ai generi preferiti dall'utente, migliorando il **richiamo** del sistema.

## 3.2) Risultati di Precisione e Recall

Le performance del metodo sono state valutate utilizzando le metriche **Precision** e **Recall**, che sono utilizzate per confrontare i film raccomandati con quelli effettivamente apprezzati dall'utente. I valori ottenuti sono i seguenti:

* **Precisione**: 0.67
* **Recall**: 0.06

Questi risultati sono stati rappresentati graficamente per una migliore comprensione delle prestazioni.

Immagine che contiene testo, schermata, diagramma, Rettangolo

Descrizione generata automaticamente

**Discussione dei Risultati**

I risultati mostrano che il valore di **Precision** è abbastanza alto, indicando che circa il 67% dei film raccomandati sono stati effettivamente apprezzati dall'utente. Tuttavia, il valore di **Recall** molto basso (0.06) suggerisce che il sistema non è riuscito a coprire una vasta gamma di film che potrebbero essere di interesse per l'utente. Questo potrebbe essere dovuto al fatto che l'algoritmo si concentra troppo sui film più simili a quelli già apprezzati, senza esplorare sufficientemente nuove possibilità.

In particolare, l'introduzione del bias per favorire i film già apprezzati ha sicuramente migliorato la precisione, ma potrebbe aver limitato la varietà delle raccomandazioni. Inoltre, la limitazione dei passi del Random Walk ha ristretto troppo l'esplorazione del grafo, riducendo la possibilità di scoprire film che potrebbero essere interessanti ma non immediatamente simili ai film già valutati.

## 3.3) Possibili Miglioramenti

Per migliorare i risultati, una strategia possibile sarebbe quella di integrare altre tecniche di apprendimento supervisionato o di esplorazione più ampia dei grafi, che possano bilanciare meglio la **Precision** e il **Recall**. Un'altra possibilità sarebbe quella di ottimizzare ulteriormente il processo di selezione dei film simili, utilizzando una **normalizzazione** delle similarità o un sistema di **pesatura dinamica** dei generi, in modo da considerare con maggiore attenzione le preferenze espresse dall'utente.

# Apprendimento Supervisionato

L'obiettivo principale della macroarea dell'apprendimento supervisionato è stato migliorare la capacità predittiva del sistema di raccomandazione attraverso l'utilizzo di metodi di validazione robusti. In particolare, è stata adottata la tecnica del K-Fold Cross Validation per testare l'affidabilità dei modelli e garantire una valutazione accurata delle prestazioni su diversi sottoinsiemi dei dati.

La K-Fold Cross Validation è una tecnica che suddivide il dataset in K sottoinsiemi, o "fold", utilizzando alternativamente ogni fold come set di test e il restante come set di addestramento. Questo approccio consente di ridurre il rischio di overfitting e fornisce una stima più robusta della performance del modello, in quanto ogni esempio del dataset viene utilizzato sia per l'addestramento che per il test. In questo progetto, il K-Fold è stato utilizzato per validare il modello di apprendimento supervisionato impiegato nel sistema di raccomandazione, in modo da ottimizzare le previsioni dei film da consigliare in base alle preferenze degli utenti.

## 4.1) Implementazione e Risultati del K-Fold Cross Validation

Nel contesto delle raccomandazioni cinematografiche, il modello utilizzato ha applicato due diverse metriche di similarità: la cosine similarity e la Pearson correlation. Le metriche hanno mostrato un buon livello di coerenza, con il punteggio della cosine similarity che ha raggiunto un valore di 0.912, indicando una forte correlazione tra gli utenti, mentre la Pearson correlation ha prodotto un valore di 0.321, suggerendo una correlazione più moderata ma comunque significativa.

I risultati della validazione incrociata su vari modelli hanno mostrato performance contrastanti ma comunque interessanti, evidenziando un ampio spazio per miglioramenti. La tabella seguente riporta i risultati ottenuti per diverse configurazioni di modelli basati sull'algoritmo KNN:

| Modello | RMSE | Tempo di Training (s) | Tempo di Test (s) |
| --- | --- | --- | --- |
| UserKNN-Base | 0.8947 | 0.0550 | 0.4226 |
| UserKNN-Amp0.5 | 0.8934 | 0.2396 | 0.3749 |
| UserKNN-Amp3.0 | 0.9135 | 0.2286 | 0.3658 |
| UserKNN-IDF | 0.8956 | 0.0455 | 0.4561 |
| UserKNN-BM25 | 0.8955 | 0.0564 | 0.4124 |

**Discussione dei Risultati**

I risultati mostrano che la tecnica K-Fold Cross Validation ha fornito una stima realistica delle prestazioni dei modelli. In particolare, il modello UserKNN-Base ha ottenuto il miglior valore di RMSE (Root Mean Squared Error), pari a 0.8947, indicando una buona capacità di previsione in generale. Tuttavia, la variazione dei tempi di addestramento e di test suggerisce che ottimizzare le configurazioni dei modelli potrebbe ridurre significativamente i tempi di esecuzione, migliorando l'efficienza.

Le metriche di performance, come il Precision (0.634), il Recall (0.652), e l'ROC AUC (0.919), hanno evidenziato una solida capacità del sistema di identificare film potenzialmente interessanti e di rilevare una buona parte dei film che l'utente potrebbe apprezzare (alto recall), con un valore di precisione che suggerisce una raccomandazione accurata per circa il 63% dei film suggeriti. L'alta performance in ROC AUC (0.919) suggerisce anche che il sistema è stato in grado di separare bene i film rilevanti da quelli irrilevanti.

Immagine che contiene testo, linea, diagramma, Diagramma

Descrizione generata automaticamente

## 4.2) Possibili Miglioramenti

Per migliorare ulteriormente i risultati, una strategia possibile sarebbe quella di ottimizzare le tecniche di selezione dei parametri, come la ricerca a griglia (grid search), per affinare i modelli KNN, nonché esplorare l'uso di modelli più avanzati, come reti neurali o ensemble methods, per combinare i punti di forza di più approcci. Inoltre, sarebbe utile esplorare vari metodi di pesatura delle caratteristiche degli utenti, per ottenere previsioni più personalizzate e accurate.