STUDENTI:

Citarella Emanuele (mat. 697086) - e.citarella1@studenti.uniba.it

Gadaleta Alessia (mat. 697885) – a.gadaleta27@studenti.uniba.it

Iacobellis Giorgia (mat. 696994) - g.iacobellis27@studenti.uniba.it

Ladisa Mattia Sebastiano (mat. 697887) – m.ladisa19@studenti.uniba.it

Relazione Caso di Studio- Ingegneria della Conoscenza

Utilizzo di tecniche di apprendimento supervisionato per classificazione di film e serie TV in base al genere, recommender system basato su clustering e costruzione di una base di conoscenza

Repository: https://github.com/giorgiaiacobellis/Icon_2020-2021.git

1. *INTRODUZIONE*

L'obiettivo del caso di studio è quello di sfruttare informazioni relative a film e serie tv presenti sulla piattaforma Netflix al fine di realizzare un sistema di classificazione e raccomandazione di film e serie tv stesse.

Nello specifico, il progetto è suddiviso in quattro sezioni principali:

- *Preprocessing* , finalizzato all'adattamento dei dati per renderli più conformi all'utilizzo successivo;
- *Classificazione*, sfruttando tecniche di apprendimento supervisionato con confronto e valutazione di diversi classificatori e relative performance, individuando il classificatore più performante per gli obiettivi determinati e seguente utilizzo di esso per predizione del genere di un film fornito dall'utente;
- *Recommender system*, utilizzato per suggerire all'utente film simili ad uno fornito da egli stesso, sfruttando alla base la tecnica di apprendimento non supervisionato del clustering;
- *Base di conoscenza*, finalizzata a consentire all'utente di effettuare domande sulle conseguenze logiche e ricevere risposte dalla macchina stessa.

Strumenti:

Per la realizzazione è stato utilizzato il linguaggio Python, scelto poichè particolarmente performante nello sviluppo di sistemi basati su conoscenza e per la quantità di librerie disponibili e utili agli obiettivi del progetto.

Tutte le librerie e versioni utilizzate sono definite nel file "*requirements.txt*".

In particolare le librerie utilizzate sono state:

- *Pandas*, libreria utile per la manipolazione e l'analisi dei dati, utilizzata nella sezione relativa al preprocessing;

- *scikit-learn*, libreria utile per le tecniche di apprendimento, utilizzata nella sezione relativa alla classificazione e alla clusterizzazione;
- *matplotlib*, libreria utile alla realizzazione di grafici, utilizzata nella sezione relativa alla classificazione e alla clusterizzazione;
- *imb-learn,* libreria utile a gestire set di dati sbilanciati, utilizzata nella sezione relativa alla classificazione;
- *kmodes*, libreria utile a realizzare clusterizzazione mediante algoritmo K-Modes, utilizzata nella sezione relativa la clusterizzazione stessa;
- *fuzzywuzzy*, libreria utile a calcolare similarità tra stringhe, utilizzata nella sezione relativa la clusterizzazione;
- *numpy*, libreria utile per eseguire calcoli su vettori e matrici, utilizzata in tutte le sezioni del progetto.

Per il testing e la cross validation dei classificatori è stata utilizzata la piattaforma *Google Colaboratory*, sfruttando la potenza di calcolo messa a disposizione da essa per velocizzare il processo.

2. PREPROCESSING DEI DATI

I dataset utilizzati nel caso di studio sono stati reperiti dal sito *Kaggle* in formato *csv* e sono i seguenti:

- Dataset film Netflix (2_Netflix_Movie.csv)
- Dataset film e serie tv Netflix (1_Netfix_Movie.csv)
- Dataset IMDB ratings film (IMDb_rating.csv)

Le informazioni fornite dai dataset sono presentate nelle seguenti tabelle:

1_Netflix_Movie

type	title	director	cast	country	release_year	duration	qenre	description
TV Show	3%	nan	João Miguel, Bianca Co…	Brazil	2020	4 Seasons	International TV S…	In a future where the elite inhabit an isla…
Movie	7:19	Jorge Michel Grau	Demián Bichir, Héctor …	Mexico	2016	93 min	Dramas, Internatio…	After a devastating earthquake hits Mexico …
Movie	23:59	Gilbert Chan	Tedd Chan, Stella Chun…	Singapore	2011	78 min	Horror Movies, Int…	When an army recruit is found dead, his fel…
Movie	9	Shane Acker	Elijah Wood, John C. R…	United States	2009	80 min	Action & Adventure…	In a postapocalyptic world, rag-doll robots…
Movie	21	Robert Luketic	Jim Sturgess, Kevin Sp…	United States	2008	123 min	Dramas	A brilliant group of students become card-c…
TV Show	46	Serdar Akar	Erdal Beşikçioğlu, Yas…	Turkey	2016	1 Season	International TV S…	A genetics professor experiments with a tre…
Movie	122	Yasir Al Yasiri	Amina Khalil, Ahmed Da…	Egypt	2019	95 min	Horror Movies, Int…	After an awful accident, a couple admitted …
Movie	187	Kevin Reynolds	Samuel L. Jackson, Joh…	United States	1997	119 min	Dramas	After one of his high school students attac…
Movie	706	Shravan Kumar	Divya Dutta, Atul Kulk…	India	2019	118 min	Horror Movies, Int…	When a doctor goes missing, his psychiatris…
Movie	1920	Vikram Bhatt	Rajneesh Duggal, Adah …	India	2008	143 min	Horror Movies, Int…	An architect and his wife move into a castl…
Movie	1922	Zak Hilditch	Thomas Jane, Molly Par…	United States	2017	103 min	Dramas, Thrillers	A farmer pens a confession admitting to his…
TV Show	1983	nan	Robert Więckiewicz, Ma…	Poland, United States	2018	1 Season	Crime TV Shows, In…	In this dark alt-history thriller, a naïve …
TV Show	1994	Diego Enrique Osorno	nan	Mexico	2019	1 Season	Crime TV Shows, Do…	Archival video and new interviews examine M…
Movie	2,215	Nottapon Boonprakob	Artiwara Kongmalai	Thailand	2018	89 min	Documentaries, Int…	This intimate documentary follows rock star…
Movie	3022	John Suits	Omar Epps, Kate Walsh,…	United States	2019	91 min	Independent Movies…	Stranded when the Earth is suddenly destroy…
Movie	Oct-01	Kunle Afolayan	Sadiq Daba, David Bail…	Nigeria	2014	149 min	Dramas, Internatio…	Against the backdrop of Nigeria's looming i…
TV Show	Feb-09	nan	Shahd El Yaseen, Shail…	nan	2018	1 Season	International TV Shows, TV Dramas	As a psychology professor faces Alzheimer's…

2_Netflix_Movie

title	duration	release_year	genre	director	cast	country	rating
#FriendButMarried	102	2018	Dramas, International Movies, Romantic Movies	Rako Prijanto	Adipati Dolken, Vanesha…	Indonesia	7
#Selfie	125	2014	Comedies, Dramas, International Movies	Cristina Jacob	Flavia Hojda, Crina Sem…	Romania	6.1
#Selfie 69	119	2016	Comedies, Dramas, International Movies	Cristina Jacob	Maia Morgenstern, Olimp…	Romania	6.3
#realityhigh	99	2017	Comedies	Fernando Lebrija	Nesta Cooper, Kate Wals…	United States	5.2
10 Days in Sun City	87	2017	Comedies, International Movies, Romantic Movies	Adze Ugah	Ayo Makun, Adesua Etomi…	South Africa	5.3
10 jours en or	97	2012	Comedies, Dramas, International Movies	Nicolas Brossette	Franck Dubosc, Claude R…	France	5.8
100 Meters	109	2016	Dramas, International Movies, Sports Movies	Marcel Barrena	Dani Rovira, Karra Elej…	Portugal, Spain	7.5
1000 Rupee Note	89	2014	Dramas, International Movies	Shrihari Sathe	Usha Naik, Sandeep Path…	India	7.3
12 ROUND GUN	90	2017	Dramas, Independent Movies, Sports Movies	Sam Upton	Sam Upton, Jared Abraha…	United States	4.7
122	95	2019	Horror Movies, International Movies	Yasir Al Yasiri	Amina Khalil, Ahmed Daw…	Egypt	7.1
13 Cameras	90	2015	Horror Movies, Indepen…	Victor Zarcoff	PJ McCabe, Brianne Monc…	United States	5.1
13 Sins	93	2014	Horror Movies, Thrillers	Daniel Stamm	Mark Webber, Rutina Wes…	United States	6.3
14 Blades	113	2010	Action & Adventure, International Movies	Daniel Lee	Donnie Yen, Zhao Wei, W…	Hong Kong, China, Singapore	6.3
14 Cameras	89	2018	Horror Movies, Thrillers	Scott Hussion, Seth Fuller	Neville Archambault, Am…	United States	4.5
15 August	124	2019	Comedies, Dramas, Independent Movies	Swapnaneel Jayakar	Rahul Pethe, Mrunmayee …	India	5.8
18 Presents	114	2020	Dramas, Independent Mo…	Francesco Amato	Vittoria Puccini, Bened…	Italy	6.7
1898: Our Last Men in the Philippines	130	2016	Dramas, International Movies	Salvador Calvo	Luis Tosar, Javier Guti…	Spain	6.5
1920	143	2008	Horror Movies, Interna…	Vikram Bhatt	Rajneesh Duggal, Adah S…	India	6.4
1922	103	2017	Dramas, Thrillers	Zak Hilditch	Thomas Jane, Molly Park…	United States	6.3
2 Alone in Paris	97	2008	Comedies, International Movies	Ramzy Bedia, Éric Judor	Ramzy Bedia, Éric Judor…	France	5.4
2 States	143	2014	Comedies, Dramas, International Movies	Abhishek Varman	Alia Bhatt, Arjun Kapoo…	India	6.9

IMDb_rating

index	title	total_votes	mean_vote
0	Miss Jerry	154	5.9
1	The Story of the Kelly Gang	589	6.3
2	Den sorte drøm	188	6
3	Cleopatra	446	5.3
4	L'Inferno	2237	6.9
5	From the Manger to the Cross; or, Jesus of Nazareth	484	5.8
6	Madame DuBarry	753	6.8
7	Quo Vadis?	273	6.2
8	Independenta Romaniei	198	7.1
9	Richard III	225	5.4
10	Atlantis	331	6.6
11	Fantômas - À l'ombre de la guillotine	1944	6.6
12	Il calvario di una madre	948	7.2
13	Juve contre Fantômas	1349	6.5
14	Ma l'amor mio non muore...	100	6.3
15	Maudite soit la guerre	124	6.7
16	Le mort qui tue	1050	6.6
17	Amore di madre	187	6.1
18	Lo studente di Praga	1768	6.5
19	Traffic in Souls	552	6
20	Gli ultimi giorni di Pompei	474	6.1

Per rendere i dati adatti e conformi alle operazioni da svolgere successivamente, sono state effettuate diverse operazioni di preprocessing, ossia:

- Unificazione dei tre dataset tramite merge per ottenere uno unico finale;
- Eliminazione delle colonne ritenute superflue ai fini del progetto;
- Rimozione dei duplicati;
- Discretizzazione della colonna ***year***, sostituendola con la colonna ***year_range*** ;
- Riduzione dei generi associati a ciascun film, mantenendone uno unico per ciascuno;
- Standardizzazione della colonna ***duration***, a causa di discordanze dell'unità di misura utilizzata nei diversi dataset;
- Riduzione degli attori presenti nella colonna ***cast***, mantenendone uno unico per ciascun film, effettuando la scelta sulla base delle occorrenze degli attori stessi nel dataset e optando per quelli che risultano maggiormente citati;
- Inserimento del valore 'Movie' nella colonna ***type*** per le row che presentavano un valore nullo ma proveniente dal dataset contenente unicamente film e non serie tv;
- Conversione dei valori della colonna ***genres*** da categorici a numerici mediante metodo di conversione delle *dummy variables*, utile per la successiva operazione di *imputation;*
- Ridenominazione dei valori nella colonna ***genres***, per renderli coerenti tra loro;
- Conversione dei valori nella colonna ***type***, da categorici a numerici mediante tecnica di conversione del *label encoder,* utile per la successiva operazione di *imputation*;
- Conversione dei valori nella colonna ***year_range***, da categorici a numerici mediante tecnica di conversione del *label encoder,* utile per la successiva operazione di *imputation*;
- Conversione dei valori nella colonna ***director***, da categorici a numerici mediante tecnica di conversione del *label encoder,* utile per la successiva operazione di *imputation*;
- Conversione dei valori nella colonna ***title***, da categorici a numerici mediante tecnica di conversione del *label encoder,* utile per la successiva operazione di *imputation*;
- Riduzione dei valori presenti nella colonna ***country***, mantenendone uno unico per ciascun film e conversione degli stessi da categorici a numeri mediante tecnica di conversione del *label encoder*;
- *Feature imputation* per i valori della colonna ***ratings*** mancanti tramite KNNImputer;
- *Feature imputation* per i valori della colonna ***genre*** mancanti tramite *hot-deck imputation;*
- Eliminazione delle row con informazioni mancanti su cui l'operazione di values imputation era impossibile da effettuare;
- Standardizzazione dei valori della colonna ***ratings***.

Il dataset ottenuto viene poi utilizzato per classificazione e clustering, in cui subisce ulteriori piccole modifiche per renderlo adatto alle funzioni da eseguire.

3. *CLASSIFICAZIONE*

Classificazione e predizione sono processi che consistono nel creare modelli che possono essere usati per descrivere degli insiemi di dati o per fare predizioni future.

Il processo di classificazione può essere visto come un processo a tre fasi: addestramento, in cui si produce un modello da un insieme di dati detto training set, stima dell'accuratezza, in cui si stima l'accuratezza del modello usando un insieme di test e utilizzo del modello, in cui si classificano istanze di classe ignota.

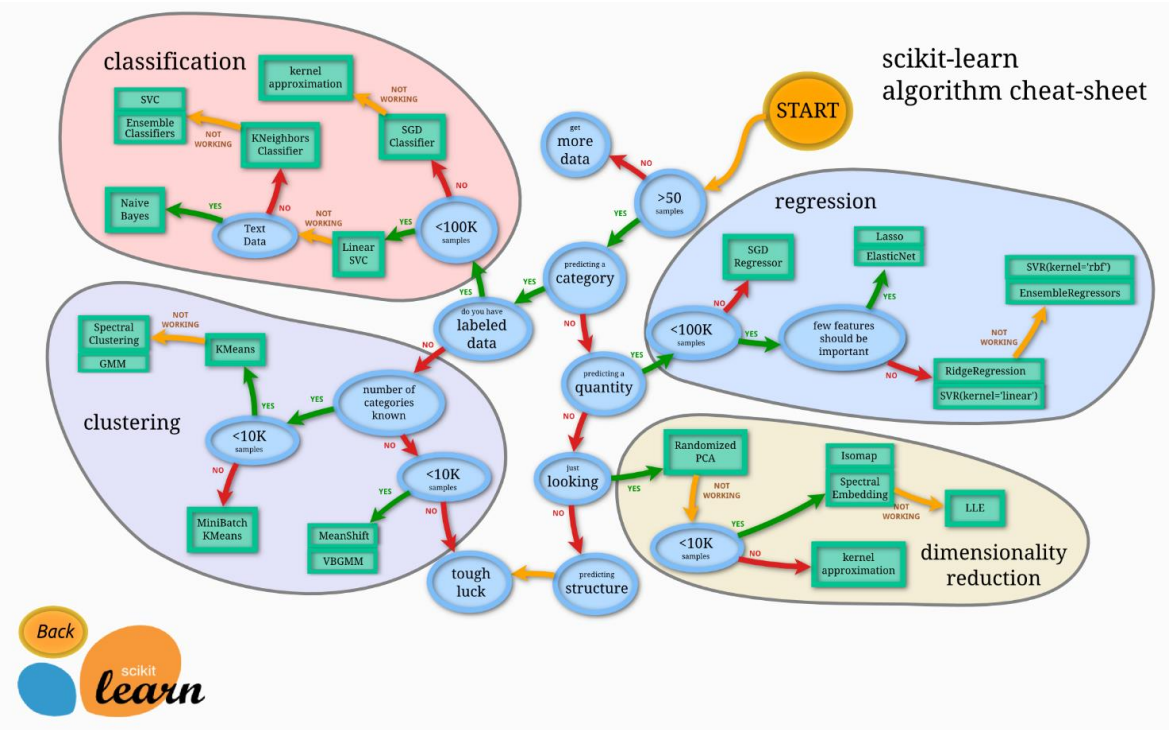
La classificazione nel caso di studio è stata utilizzata con lo scopo di predire, tramite addestramento sul dataset precedentemente ottenuto, il genere di un film fornito dall'utente.

La feature target, ossia la label da predire, è quindi relativa la colonna ***genre***.

Il dataset utilizzato è abbastanza complesso e ampio, pertanto è stata necessaria una prima fase dedita ad una ricerca accurata del classificatore più adatto, ossia quello in in grado di gestire il dataset in questione. Infatti, nel machine learning esiste il teorema "*No free Lunch*" che afferma che non esiste un algoritmo che vada bene per qualsiasi problema e, di conseguenza, sono necessarie varie prove per trovare il modello di predizione più accurato, valutando le performance di ciascuna alternativa, al fine di trovare la più adatta.

Il metodo utilizzato per attuare questa ricerca è stato quello di considerare i classificatori le cui caratteristiche sembravano ottimali per il caso di studio, per poi valutarne le performance tramite cross validation ed in seguito effettuare un confronto tra tutti.

Una volta trovato il più performante, questo è stato utilizzato per le predizioni del sistema.



La scelta degli algoritmi da testare si è, in parte, basata sulla guida grafica presente nell'immagine. (fonte https://scikit-learn.org/stable/tutorial/machine_learning_map/).

I classificatori considerati sono:

- **K-Nearest Neighbors Classifier (KNN)**

Questo classificatore restituisce come output il genere di appartenenza del film dato in input, basando la classificazione sulla pluralità dei voti dei suoi vicini, cioè viene assegnata la classe più presente tra i k film più simili ritrovati, calcolati per similarità dal film da definire dato in input – per determinare il k, uno dei metodi maggiormente impiegati grazie alla sua efficienza è la *cross validation*, mentre riguardo i calcoli della distanza, il metodo prevede che i film siano rappresentati come vettori di posizione in uno spazio multidimensionale.

È un tipo di classificatore non generalizzante, poiché attua predizioni ricordando i dati di addestramento, piuttosto che costruendo un nuovo modello.

È la tecnica **più semplice** che si può applicare, **spesso efficace** ma **lenta** e richiede **molta memoria** poichè il costo di calcolo è quadratico.

- **C-Support Vector Classification (SVC)**

L'algoritmo SVM, seppur utilizzato per classificazioni binarie, può essere impiegato per problemi di classificazione multiclasse, utilizzando la metodologia one-vs-one.

Nello specifico, si creano k(k-1)/2 classificatori, dove k è il numero di classi, che effetuano classificazione su coppie di classi, per poi assegnare come classe finale quella con più assegnazioni.

L'SVM è basato sull'idea di trovare un iperpiano che divida al meglio un set di dati in due o più classi su x dimensioni spaziali dove x è il numero di classi. I punti dati più vicini all'iperpiano sono detti *vettori di supporto* e sono i vettori rappresentativi delle possibili classi di appartenenza.

Un iperpiano linearmente separabile è un iperpiano cui è semplice distinguere due classi, il problema è trovare quale tra le infinite rette che rappresentano l'iperpiano risulti ottimale, ossia quella che generi il minimo errore di classificazione su una nuova osservazione.

Il **metodo del kernel** ci consente di modellare modelli non lineari di dimensioni superiori. Il suo scopo è di prendere i dati come input e trasformarli nella forma richiesta qualora non sia possibile determinare un iperpiano linearmente separabile.

Nel nostro caso useremo il metodo kernel con l'SVM poichè abbiamo un modello non lineare e lo testeremo su vari kernel per trovare il più adatto.

I principali vantaggi di questo algoritmo sono i seguenti:

- **Efficace in dimensioni spaziali elevate**
- **Efficienza della memoria**, poiché solo un sottoinsieme dei punti di allenamento viene utilizzato nel processo decisionale effettivo di assegnazione
- **Versatilità**, grazie alla capacità di applicare nuovi kernel portando a una maggiore performance di classificazione.

Tra gli svantaggi principali abbiamo:

- **Interpretazione non semplice**, e quindi la mancanza di trasparenza dei risultati.
- **Metodo non probabilistico**, poiché il classificatore funziona posizionando gli oggetti sopra e sotto un iperpiano di classificazione, non esiste un'interpretazione probabilistica diretta per l'appartenenza al gruppo

-**Bagging classifier**

Il *bagging* si basa sull'addestrare più modelli dello stesso tipo, ciascuno su sottoinsiemi casuali del dataset originale e quindi aggrega le loro previsioni individuali (mediante voto o media) per formare una previsione finale.

Ogni *weak learner* viene addestrato in parallelo con un set di addestramento che viene generato estraendo casualmente, con sostituzione, N esempi (o dati) dal dataset originale (dove N è la dimensione del dataset). Il training set per ciascuno dei classificatori di base è indipendente l'uno dall'altro.

Il bagging viene usato soprattutto quando l'obiettivo è ridurre la varianza(*overfitting*) del classificatore, in modo da evitare che si abbia un'ottima precisione sui dati di addestramento e alte percentuali di errore sui dati di test.

Gli stimatori maggiormente considerati sono gli alberi di decisione, definiti in molti casi come base learner del bagging classifier.

-Random Forest Classifier

La *Random Forest* costruisce un insieme di alberi decisionali, uniti per ottenere una previsione più accurata e stabile.

Ogni albero in una random forest impara da un campione casuale di dati. I campioni vengono disegnati con la sostituzione, nota come *bootstrap*, il che significa che alcuni campioni verranno utilizzati più volte in un singolo albero.

L'idea è che addestrando ciascun albero su campioni diversi, sebbene ogni albero possa presentare una varianza elevata rispetto a una particolare serie di dati di addestramento, nel complesso l'intera foresta avrà una varianza inferiore, in modo da avere le predizioni finali vicine al risultato

I vantaggi del Random Forest sono la **versatilità**, perché può essere utilizzato sia per problemi di regressione che di classificazione, funzionando bene con una combinazione di caratteristiche numeriche e categoriche, e l'avere gli **iperparametri predefiniti ottimali**, perchè producono un buon risultato di previsione, consentendo anche un notevole miglioramento di essi, e di conseguenza della previsione.

In generale, questi algoritmi sono **veloci** da **addestrare**, ma piuttosto **lenti** nel creare **previsioni** una volta che sono stati addestrati, poiché richiedono un alto numero di alberi per ottenere risultati accurati.

<u>TUNING DEGLI IPERPARAMETRI</u>

Per trovare il classificatore più performante è stato necessario anche ricercare i paramentri migliori per il caso. Gli iperparametri sono parametri che non vengono appresi direttamente all'interno dei classificatori ma devono essere forniti prima dell'apprendimento. È possibile e consigliato cercare nello spazio iperparametrico il miglior punteggio di cross validation per determinarli. Qualsiasi parametro fornito durante la costruzione di uno stimatore può essere ottimizzato in questo modo.

La ricerca degli iperparametri è stata fatta tramite l'uso di *GridSearchCV*, fornito dalla libreria *scikit-learn*, che genera in modo esaustivo candidati da una griglia di valori dei parametri specificati.

Il GridSearchCV implementa la *API estimator*: quando la si "adatta" su un set di dati vengono valutate tutte le possibili combinazioni di valori dei parametri e viene mantenuta la combinazione migliore. Successivamente si effettua di nuovo il training con il metodo e i parametri migliori su tutti i dati e si ottiene modello finale.

CROSS VALIDATION

Per evitare l'overfitting,ossia un legame eccessivo del modello ai dati che non permette di generalizzare ed effettuare correttamente la classificazione, è pratica comune quando si esegue un esperimento di apprendimento automatico tenere parte dei dati disponibili come set di test.

La procedura più comune utilizzata a questo scopo è chiamata *cross validation* (CV in breve). Nell'approccio di base, chiamato *k- fold CV*, il training set è suddiviso in k insiemi più piccoli. Per ciascuno del k sottoinsiemi si segue la seguente procedura :

1. Un modello viene addestrato utilizzando k-1 sottoinsiemi come dati di allenamento;
2. il modello risultante viene convalidato sulla parte restante dei dati

La misura delle prestazioni riportata dalla convalida incrociata k- fold è quindi la media dei valori calcolati nel ciclo.

Come scegliere il parametro k dipende dal tempo e dalle risorse disponibili. I valori usuali per k sono 3, 5, 10 o anche N, dove N è la dimensione dei dati.

RISULTATI OTTENUTI:

I modelli di classificazione devono essere valutati per determinare il loro grado di efficienza del compiere un task specifico. È importante avere classificatori con una buona performance perché sono utili in fase di predizione e, di conseguenza, permettono di ottenere predizioni ottimali.

Le metriche analizzate nella ricerca del classificatore migliore sono state:

- Precision,

$$\frac{tp}{tp + fp}$$

  è il rapporto tra le istanze ritrovate e definite come rilevanti e le istanze ritrovate;

- Recall,

$$\frac{tp}{tp + fn}$$

  è il rapporto tra le istanze ritrovate e definite come rilevanti e le istanze effettivamente rilevanti;

- Accuratezza,

$$\frac{(\mathbf{TP} + \mathbf{TN})}{(\mathbf{TP} + \mathbf{FP} + \mathbf{TN} + \mathbf{FN})}$$

  è il rapporto tra le istanze correttamente predette e il totale delle istanze presenti. Testa l'abilità del modello nel predire correttamente le due classi;

- F1-Score,

$$F = \frac{2PR}{P+R} = \frac{2}{\frac{1}{R}+\frac{1}{P}}$$

è un modo per combinare la precisione e il richiamo, infatti è la loro media armonica. Tiene conto sia dei falsi positivi che dei falsi negativi.

Tramite la Cross Validation, abbiamo effettuato il tuning degli iperparametri per ogni classificatore analizzato, basandoci sulle metriche precedenti quali precisione, richiamo e accuratezza.

Per ciascuno dei classificatori, abbiamo ottenuto il sottoinsieme con gli iperparametri migliori.

Inoltre, abbiamo testato ciascuno di essi per valutare le performance di classificazione per ogni genere presente nel dataset e come risultato di questi test abbiamo ottenuto un report con informazioni relative alle metriche utilizzate.

- Esempio di report per l'accuratezza per il KNN Classifier:

```
# Tuning degli iperparametri per la metrica accuracy

Miglior combinazione di parametri ritrovata:

{'metric': 'manhattan', 'n_neighbors': 1, 'weights': 'uniform'}
Classification report:

Il modello è stato addestrato sul training set completo


 Le metriche sono state calcolate sul test set.

              precision    recall  f1-score   support

       anime       0.99      1.00      1.00       360
        cult       0.99      1.00      1.00       360
     fantasy       0.98      1.00      0.99       380
      action       0.90      1.00      0.95       342
 documentary       0.88      0.96      0.92       346
      nature       0.93      1.00      0.96       360
    romantic       0.94      0.99      0.96       344
       sport       0.75      0.81      0.78       352
    thrillers       0.78      0.69      0.74       377
        kids       0.93      0.99      0.96       363
      dramas       0.54      0.27      0.36       351
      horror       0.95      1.00      0.97       377
     standup       0.93      0.96      0.95       356
   commedies       0.75      0.72      0.73       355
     musical       0.88      0.94      0.91       382

    accuracy                           0.89      5405
   macro avg       0.88      0.89      0.88      5405
weighted avg       0.88      0.89      0.88      5405
```

I report riguardo precisione, richiamo e accuratezza degli altri classificatori utilizzati sono presenti nella cartella "*classification_results*".

Inoltre, per ogni test abbiamo ottenuto la media dei valori precisione, richiamo, accuratezza e F1-Score per ogni classificatore:

KNN CLASSIFIER:
con iperparametri {'metric': 'manhattan', 'n_neighbors': 1, 'weights': 'uniform'}

KNN Classifier	Precision	Recall	F1-Score
accuracy			0,89
macro avg	0,88	0,89	0,88
weighted avg	0,88	0,89	0,89

BAGGING CLASSIFIER:
con iperparametri {'n_estimators': 10}

Bagging Classifier	Precision	Recall	F1-Score
accuracy			0,89
macro avg	0,87	0,88	0,87
weighted avg	0,87	0,89	0,87

RANDOM FOREST:
con iperparametri {'max_features': 'sqrt', 'n_estimators': 1000}

Random Forest Classifier	Precision	Recall	F1-Score
accuracy			0,89
macro avg	0,88	0,89	0,88
weighted avg	0,88	0,89	0,88

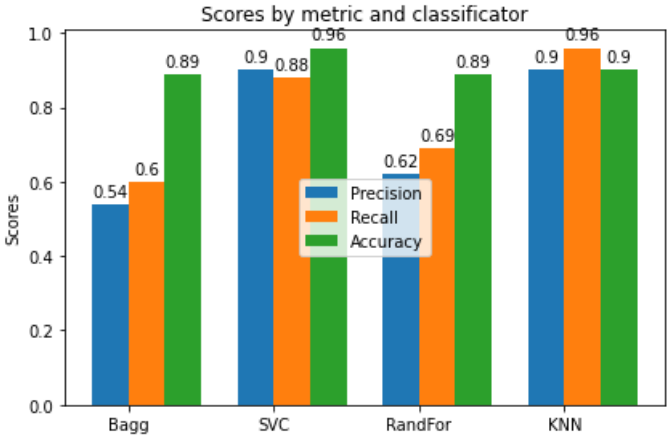
SUPPORT VECTOR MACHINE:
con iperparametri {'C': 1.0, 'gamma': 'auto', 'kernel': 'rbf', 'probability': True}

SVC Classifier	Precision	Recall	F1-Score
accuracy			0,92
macro avg	0,96	0,92	0,93
weighted avg	0,96	0,92	0,93

Considerando tutte le combinazioni di iperparametri migliori ritrovate, è stata poi effettuata una cross validation per mettere a confronto i vari algoritmi e definirne il migliore, andando a calcolare accuratezza, precisione e richiamo.

I confronti effettuati hanno evidenziato che l'algoritmo SVC e il K-Nearest-Neighbor sono quelli che risultano essere i più adatti al caso di studio. Nonostante l'SVC risulti essere migliore in precisione e accuratezza rispetto al KNN, con una differenza minima, si è preferito utilizzare come classificatore finale per il caso di studio il KNN poichè l'SVC ha dei tempi di esecuzione molto elevati, quindi si è optato per un algoritmo efficiente e rapido.

	Bagging Classifier	SVC	Random Forest	KNearestNeighbor	Best Score
Accuracy	0.889364	0.959715	0.893286	0.903030	SVC
Precision	0.541302	0.901923	0.623179	0.901229	SVC
Recall	0.602938	0.884289	0.689164	0.960617	KNearestNeighbor

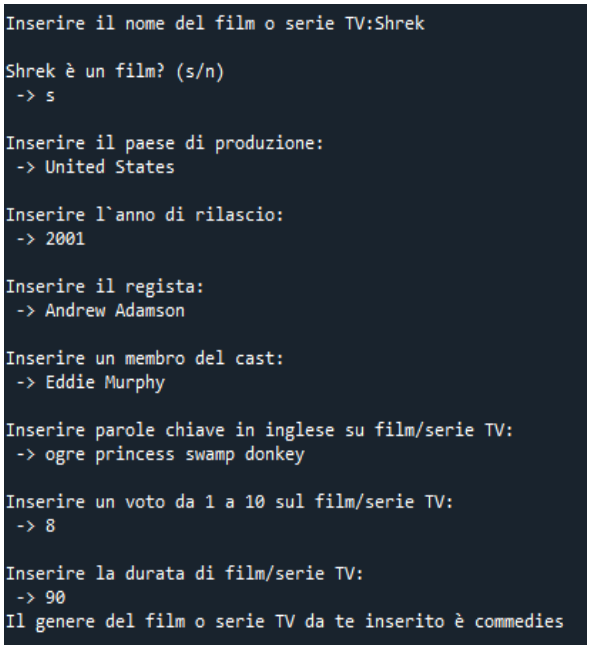


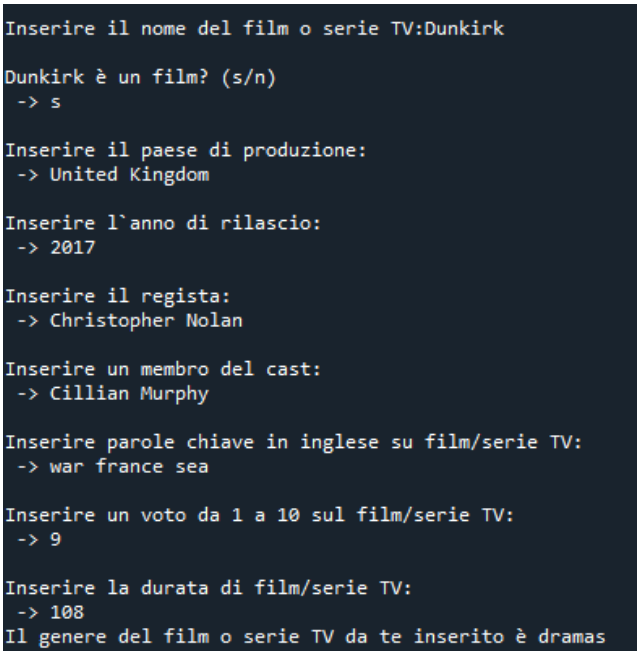
E' stato quindi addestrato il modello con il K-Nearest-Neighbor utilizzando tutto il dataset per poter essere utilizzato nella classificazione di nuovi film.

Di seguito sono riportati alcuni esempi di utilizzo del sistema:

```
Inserire il nome del film o serie TV:Mamma mia

Mamma mia è un film? (s/n)
 -> s

Inserire il paese di produzione:
 -> United States

Inserire l`anno di rilascio:
 -> 2008

Inserire il regista:
 -> Phyllida Lloyd

Inserire un membro del cast:
 -> Meryl Streep

Inserire parole chiave in inglese su film/serie TV:
 -> wedding greece fathers

Inserire un voto da 1 a 10 sul film/serie TV:
 -> 9

Inserire la durata di film/serie TV:
 -> 108
Il genere del film o serie TV da te inserito è musical
```

```
Inserire il nome del film o serie TV:Gotham

Gotham è un film? (s/n)
 -> n

Inserire il paese di produzione:
 -> United States

Inserire l`anno di rilascio:
 -> 2014

Inserire il regista:
 -> Danny Cannon

Inserire un membro del cast:
 -> Ben McKenzie

Inserire parole chiave in inglese su film/serie TV:
 -> batman mafia joker police

Inserire un voto da 1 a 10 sul film/serie TV:
 -> 8

Inserire la durata di film/serie TV:
 -> 5
Il genere del film o serie TV da te inserito è dramas
```





4. *CLUSTERING E RECOMMENDATION*

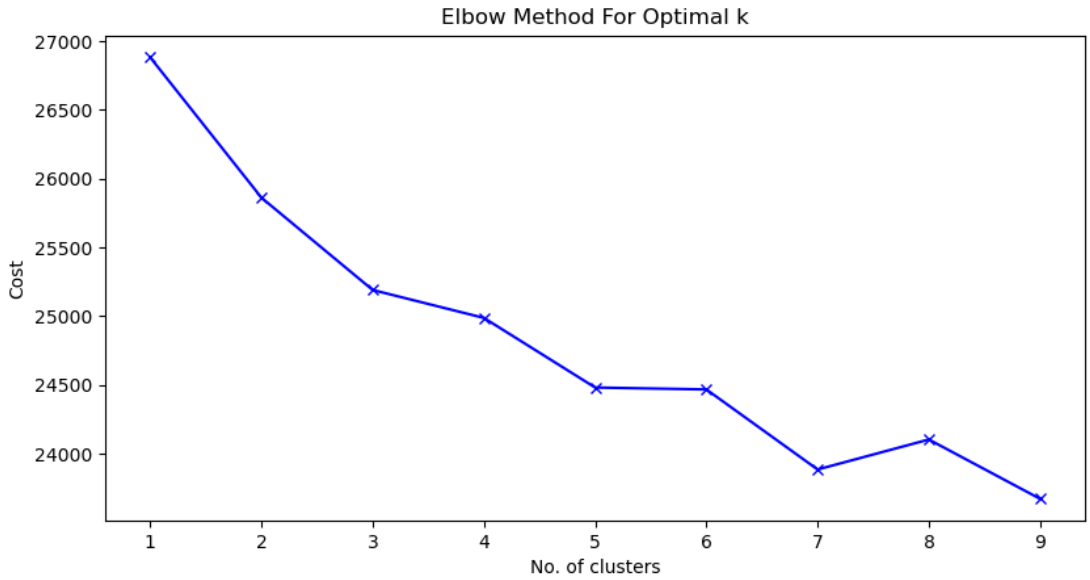
Il clustering è una metodologia di apprendimento non supervisionato che consente di identificare e raggruppare elementi simili appartenenti a dataset di grandi dimensioni, creando cluster ossia gruppi di questi ultimi che risultano conformi ad elementi medi, detti centroidi.

Nello specifico, si è scelto di adottare questa tecnica al fine di poter individuare delle nuove similarità e correlazioni tra i dati che non dipendessero unicamente dal genere dei film interessati, per poterle poi sfruttare alla base di un recommender system che si discostasse dai risultati ordinari.

a. *Clustering*

Inizialmente, si è optato per l'utilizzo dell'algoritmo *K-Means* che, però, si è scontrato con la natura del nostro dataset. Infatti, essendo più adatto a features di tipo continuo, nonostante avessimo attuato una conversione dei dati categorici con le diverse tecniche disponibili, i risultati ottenuti non hanno soddisfatto le aspettative previste poiché i cluster risultavano estremamente imprecisi e il loro contenuto poco coerente.
Si sono, quindi, cercati nuovi algoritmi che risultassero più conformi alle necessità dettate dagli obiettivi del progetto, considerando la forte presenza di variabili categoriche nel dataset, per poi optare per la tecnica K-Modes.
Infatti, questo algoritmo estende il K-Means utilizzando una misura di similarità dedita ad elementi categorici, sostituendo l'utilizzo della media con l'utilizzo della moda ed utilizzando un metodo frequency-based utile per minimizzare la funzione di costo.

Si è scelto di individuare 3 cluster, e quindi centroidi, sfruttando il ***'metodo del gomito'***, ossia un metodo empirico utile a trovare il numero ottimale di cluster per un set di dati all'interno di un range determinato – nello specifico, il range scelto è rimasto limitato al di sotto del numero di generi dei film presenti, in modo tale da poter individuare correlazioni non fortemente legate a questi.



Riportiamo qui di seguito gli head dei nostri cluster.

CLUSTER 1

type	title	director	cast	genre	country	year_range	description	ratings_range
Movie	The Informant!	Steven Soderbergh	Matt Damon	dramas	United States	2005-2010	In the bustling center of Hong K…	>5
Movie	Krishna Cottage	Santram Varma	Sohail Khan	horror	India	2000-2005	When a tough-minded ex-drug deal…	>5
Movie	The Great Raid	John Dahl	Benjamin Bratt	dramas	United States	2000-2005	When three teen outcasts arrive …	>5
Movie	The Pursuit of Happyness	Gabriele Muccino	Will Smith	dramas	United States	2005-2010	Psychic Hiroyuki Ehara leads var…	>5
Movie	The Bucket List	Rob Reiner	Jack Nicholson	dramas	United States	2005-2010	Didem tries everything to get ac…	>5
Movie	The Longshots	Fred Durst	Ice Cube	dramas	United States	2005-2010	In this fun, fast-paced music co…	>5
Movie	Poseidon	Wolfgang Petersen	Josh Lucas	dramas	United States	2005-2010	The real Mitt Romney is revealed…	>5
Movie	The Squid and the Whale	Noah Baumbach	Jeff Daniels	dramas	United States	2000-2005	The forces of family, grief and …	>5
Movie	Well Done Abba	Shyam Benegal	Boman Irani	horror	India	2005-2010	In 1890s Malacca, Li Lan finds h…	>5

CLUSTER 2

type	title	director	cast	genre	country	year_range	description	ratings_range
Movie	Black Rose	Alexander Nevsky	Alexander Nevsky	fantasy	Russia	2010-2015	The life of cheerleade…	>5
Movie	Sisterakas	Wenn V. Deramas	Ai-Ai de las Alas	kids	Philippines	2010-2015	Complications around t…	>5
Movie	Zapped	Peter DeLuise	Zendaya	commedies	Canada	2010-2015	This biopic chronicles…	>5
Movie	Pizza, birra, faso	Israel Adrián Caetano, Bruno Stagnaro	Héctor Anglada	thrillers	Argentina	1995-2000	A nameless widow juggl…	>5
Movie	Head On	Ana Kokkinos	Alex Dimitriades	thrillers	Australia	1995-2000	Mexican stand-up comed…	>5
Movie	Miss Hokusai	Keiichi Hara	Anne Watanabe	anime	Japan	2010-2015	After a chance encount…	>5
Movie	Kath & Kimderella	Ted Emery	Jane Turner	commedies	Australia	2010-2015	Despite discouragement…	>5
Movie	Medium	Jacek Koprowicz	Władysław Kowalski	thrillers	Poland	1980-1990	A father's suicide sen…	>5
Movie	Magic Snowflake	Luc Vinciguerra	Nathan Simony	kids	France	2010-2015	In Justin's dreams, he…	>5
Movie	Back to the 90s	Yanyong Kuruaungkoul	Dan Aaron Ramnarong	musical	Thailand	2010-2015	An epidemiologist turn…	>5

CLUSTER 3

type	title	director	cast	genre	country	year_range	description	ratings_range
Movie	#realityhigh	Fernando Lebrija	Nesta Cooper	commedies	United States	2015-2020	As a grisly virus rampages a city, a…	>5
Movie	¡Ay, mi madre!	Frank Ariza	Estefanía de los Santos	commedies	Spain	2015-2020	As Ayu and Ditto finally transition …	<5
Movie	Ég man þig	Óskar Thór Axelsson	Jóhannes Haukur Jóhannesson	commedies	Iceland	2015-2020	A teenage hacker with a huge nose he…	>5
Movie	Çok Filim Hareketler Bunlar	Ozan Açıktan	Ayça Erturan	commedies	Turkey	2005-2010	This documentary celebrates the 50th…	<5
Movie	Òlòtûré	Kenneth Gyang	Beverly Osu	commedies	Nigeria	2015-2020	Two days before their final exams, t…	>5
Movie	1 Mile to You	Leif Tilden	Billy Crudup	romantic	United States	2015-2020	The slacker owner of a public bath h…	>5
Movie	12 ROUND GUN	Sam Upton	Sam Upton	sport	United States	2015-2020	Upon losing his memory, a crown prin…	<5
Movie	17 Again	Burr Steers	Zac Efron	commedies	United States	2005-2010	A pregnant teen is forced by her fam…	>5
Movie	18 Presents	Francesco Amato	Vittoria Puccini	commedies	Italy	2015-2020	Young parents-to-be Claire and Ryan …	>5
Movie	1898: Our Last Men in the Philippines	Salvador Calvo	Luis Tosar	commedies	Spain	2015-2020	After a teenage girl's perplexing su…	>5
Movie	20th Century Women	Mike Mills	Annette Bening	fantasy	United States	2015-2020	Nearing a midlife crisis, thirty-som…	>5
Movie	2307: Winter's Dream	Joey Curtis	Paul Sidhu	fantasy	United States	2015-2020	A bumbling Paris policeman is dogged…	<5
Movie	27, el club de los malditos	Nicanor Loreti	Diego Capusotto	commedies	Argentina	2015-2020	When his wife is convicted of murder…	>5

b. Recommender System

Per quanto riguarda il sistema di raccomandazione, è stato adottato un approccio basato sui contenuti, incrociando gli attributi e le descrizioni dei vari film presenti nel dataset con uno apprezzato e fornito dall'utente stesso.
Nello specifico, all'utente sono richieste informazioni inerenti il film da egli apprezzato che vengono sfruttate per individuare il cluster più simile e, in questo modo, è possibile ricavare una lista di film consigliabili all'utente sulla base della similarità tra quello fornito e quelli presenti nel cluster risultato più simile.

In particolare, a seguito della clusterizzazione e l'ottenimento dei gruppi di elementi ben distinti tra loro, per calcolare le similarità si è utilizzata la libreria sopra-citata *FuzzyWuzzy* che utilizza come metrica la *distanza di Levenshtein*, ossia una metrica in grado di misurare la differenza tra due sequenze di caratteri basandosi sul numero minimo di modifiche necessarie di un singolo carattere per trasformare la parola con quella con cui viene confrontata.
Di seguito è riportato un esempio di utilizzo:

```
Benvenuto in MovieLand!

Scegli come proseguire:
 1. Scopri il genere di un film o serie TV
 2. Lasciati suggerire un nuovo film sulla base di un altro che hai apprezzato
 --> 2
INIZIAMO!

[NB: inserire i dati dei film rispettando la dicitura ufficiale]
(es. Avengers: Infinity War-> OK  ma avengers infinity war->NO)


Inserire il nome del film o serie TV:Iron Man

Iron Man è un film? (s/n)
 -> s

Inserire il paese di produzione:
 -> United States

Inserire l`anno di rilascio:
 -> 2008

Inserire il regista:
 -> Jon Favreau

Inserire un membro del cast:
 -> Robert Downey Junior
```

```
Inserire parole chiave in inglese su film/serie TV:
 -> superheroes marvel dc

Inserire un voto da 1 a 10 sul film/serie TV:
 -> 10

Inserire la durata di film/serie TV:
 -> 126

Inserisci il genere, scegliendo tra questi:
1 action
2 anime
3 commedies
4 cult
5 documentary
6 dramas
7 fantasy
8 horror
9 kids
10 musical
11 nature
12 romantic
13 sport
14 stand-up
15 thrillers
 ->7
```

```
Ti consigliamo di guardare:

Spider-Man 3
Avengers: Infinity War
Scorpion King 5: Book of Souls
Hulk Vs.
A Boy Called Po
```

5. *BASE DI CONOSCENZA*

Una base di conoscenza è una banca dati, grazie alle cui informazioni e, quindi, alle conoscenze che sono presenti al suo interno, riesce a fornire un supporto all'utente fornendogli risposte a delle domande che vengono effettuate senza la necessità di generare i possibili mondi.
Quindi, la base di conoscenza o KB è definibile come un insieme di assiomi, cioè delle proposizioni che possono essere asserite essere vere.

La base di conoscenza viene utilizzata nel caso di studio al fine di consentire all'utente e al sistema uno scambio di domande e risposte inerenti il dominio approfondito, ossia quello dei film e delle serie tv, attuando uno scambio di informazioni.
Nello specifico, l'utente può avanzare le seguenti richieste:

- Confermare la corrisponde tra *titolo* e *genere* relativi ad un film, attraverso la funzione askGenereDaTitolo, che accetta in input entrambi i dati e restituisce in output una risposta affermativa o negativa;

askGenereDaTitolo(titolo, genere) <=> titolo_genere;

esempio di funzionamento di askGenereDaTitolo("titolo","genere")

```
Digitare il titolo del film: American Psycho
Digitare il genere del film: dramas
YES
Digitare how per la spiegazione: how
askGenereDaTitolo(American Psycho,dramas) <=> American Psycho_dramas
Digitare 'how i' specificando al posto di i il numero dell'atomo : how 1
American Psycho_dramas is  True
```

```
Digitare il titolo del film: American Psycho
Digitare il genere del film: commedies
NO
Digitare how per la spiegazione: how
askGenereDaTitolo(American Psycho,commedies) <=> American Psycho_commedies
Digitare 'how i' specificando al posto di i il numero dell'atomo : how 1
American Psycho_commedies is  False
```

- Verificare se film diversi appartengano ad uno stesso genere, mediante l'utilizzo della funzione askStessoGenere, che accetta in input i titoli dei film in questione;

  askStessoGenere(titolo1, titolo2) <=> titoli(titolo1, titolo2)
  and titolo1_primoGenere
  and titolo2_secondoGenere
  and stessoGenere(primoGenere, secondoGenere), dove *stessoGenere("genere1","genere2")* indica se i generi presenti come parametri sono o meno uguali tra loro.

esempio di funzionamento di askStessoGenere("titolo1","titolo2")

```
Digitare il titolo del primo film: American Psycho
Digitare il titolo del secondo film: American Warfighter
YES
Digitare how per la spiegazione: how
askStessoGenere(American Psycho,American Warfighter) <=> American Psycho_American Warfighter and  American Psycho_dramas
and American Warfighter_dramas and generiUguali(dramas,dramas)
Digitare 'how i' specificando in i il numero dell'atomo per ulteriori informazioni: how 1
American Psycho_dramas is True
Digitare 'how i' specificando in i il numero dell'atomo per ulteriori informazioni: how 2
American Warfighter_dramas) is  True
Digitare 'how i' specificando in i il numero dell'atomo per ulteriori informazioni: how 3
```

```
Digitare il titolo del primo film: American Psycho
Digitare il titolo del secondo film: Amy
NO
Digitare how per la spiegazione: how
askStessoGenere(American Psycho,Amy) <=> American Psycho_Amy and  American Psycho_dramas and Amy_musical and generiUguali
(dramas,musical)
Digitare 'how i' specificando in i il numero dell'atomo per ulteriori informazioni: how 1
American Psycho_dramas is True
Digitare 'how i' specificando in i il numero dell'atomo per ulteriori informazioni: how 2
Amy_musical) is  True
Digitare 'how i' specificando in i il numero dell'atomo per ulteriori informazioni: how 3
generiUguali(American Psycho,Amy) <=> dramas_musical is False
```

Per ogni query che viene eseguita, ossia ogni interrogazione posta in modo tale da sapere se una proposizione sia conseguenza logica della base di conoscenza, la KB risponderà con *YES* oppure *NO* a seconda del tipo di clausola che le viene presentata.
Inoltre, si potrà chiedere la motivazione secondo la quale si è ottenuto un determinato risultato attraverso l'operatore *how* - in questo modo, la base potrà fornire la motivazione alla base della restituzione di una certa risposta rispetto ad un'altra mostrando le clausole utilizzate per dedurre la risposta.
Infine, l'utente ha la possibilità di richiedere una prova per ogni atomo nel corpo di una clausola.