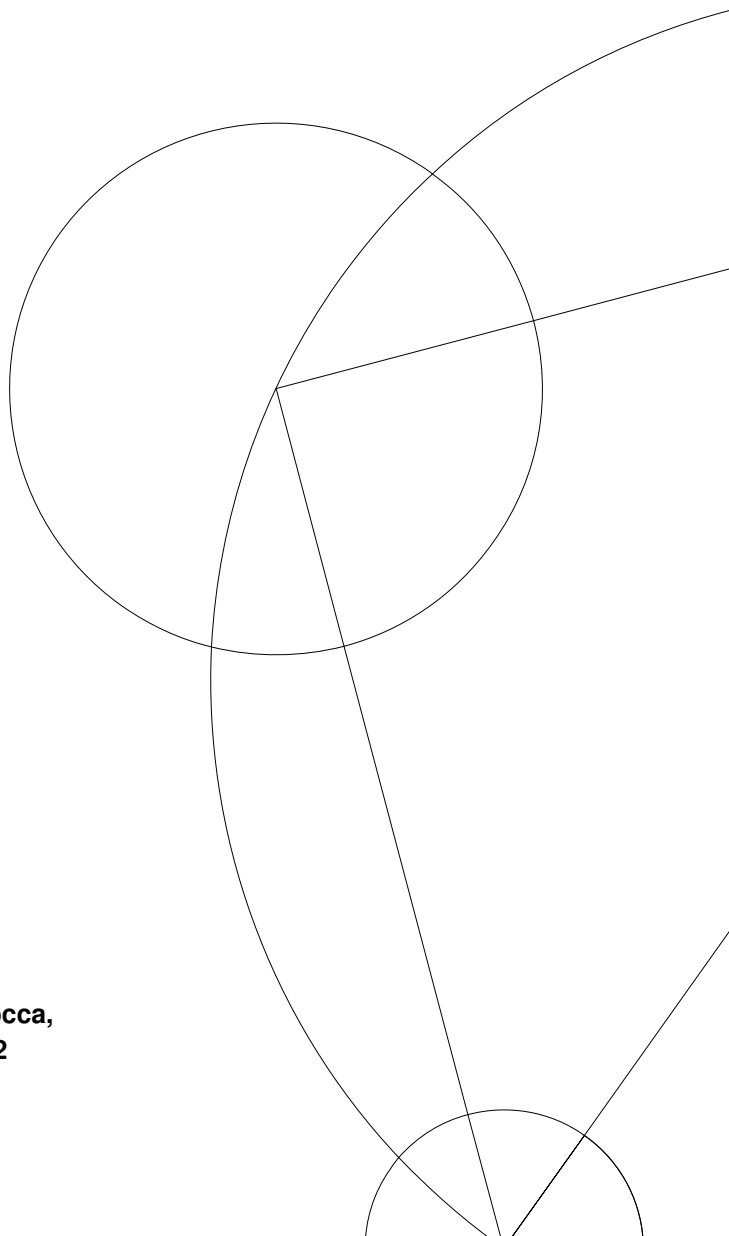
Analisi del Mondo Cinematografico tra il 2019 e il 2021

Progetto realizzato da: Emanuela Elli (892901), Federica Madon (825628), Eleonora Rossi (886756).



CdLM Data Science, Università degli studi di Milano Bicocca,
Corso di Data Management, Anno Accademico 2021-2022

Introduzione

La pandemia da Covid-19 ha influenzato le nostre vite sotto molti punti di vista. Lo scopo di questo progetto è osservare se vi è la presenza (o meno) di un cambiamento anche nel mondo cinematografico. Per tale motivo, si è deciso di considerare i dati relativi ai film prodotti, a livello internazionale, compresi nel range temporale dall'anno 2019 (l'anno precedente l'arrivo della pandemia) all'anno 2021 (entrambi compresi). Gli aspetti principali, su cui si è incentrato questo studio, sono i seguenti:

- i generi di film più visti;
- gli incassi dei film;
- i premi vinti durante la cerimonia degli Oscar.

Durante la fase di ricerca dei dati e di studio del dominio considerato, si è deciso di aggiungere, agli aspetti precedentemente elencati, un ulteriore elemento relativo al ***Test di Bechdel***, per evidenziare un aspetto sociale relativo alla presenza femminile nel mondo cinematografico. Chiaramente, anche per questo dataset, è stato considerato il range temporale 2019-2021 per avere consistenza nei dati.

Prima di descrivere le procedure effettuate durante le diverse fasi del progetto, è importante introdurre cosa sia effettivamente il ***Test di Bechdel***.

Cos'è il Test di Bechdel

Il test di Bechdel è un metodo di valutazione, che si basa su criteri specifici, con lo scopo di "misurare" la rappresentazione femminile nelle opere di finzione. I criteri su cui si fonda sono la presenza, all'interno della trama, di due personaggi femminili che parlano tra loro di argomenti che non riguardino una figura maschile.

Tale test prende il nome dalla fumettista americana Alison Bechdel, che enuncia i suddetti criteri, inizialmente per scherzo, all'interno di una vignetta del 1985 dal titolo *The Rule* attraverso il personaggio di *Mo*, una donna di colore che accetta di accompagnare l'amica Ginger al cinema solo se il film in questione soddisferà i requisiti sopracitati.

Figura 1: La vignetta con la quale Bechdel introdusse il "test".

Pertanto si può assumere che il film supera il test se:

1. presenta una scena con protagoniste almeno due donne;
2. le due donne devono parlare tra di loro;
3. l'argomento di cui parlano non deve riguardare un uomo.

È evidente come questo test non si possa considerare uno strumento rigoroso per determinare se un dato contenuto diffonda o meno stereotipi sessisti, ma sicuramente l'assenza di almeno due personaggi femminili, che abbiano argomenti di conversazione diversi da uomini, è un dato significativo per la loro rilevanza all'interno di un film. Ci sono infatti film, ad esempio, che superano il test come *Mean Girls* in cui le figure femminili, certamente presenti in abbondanza, hanno una rappresentazione estremamente stereotipata della donna. Altro esempio sono i film horror in cui le donne possono essere presenti anche in grande numero ma vengono spesso rappresentate come vittime. Allo stesso modo si può notare come supera il test il film di animazione *Cenerentola*, in cui l'immagine femminile risente ancora fortemente di linguaggi sessisti e immagini stereotipate, e non *Shrek*, sebbene la figura femminile in esso rappresentata (Fiona), donna determinata e indipendente.

Nonostante il test non misuri quanto un film sia femminista o ben fatto, può comunque essere considerato un potentissimo e valido mezzo per generare riflessioni nello spettatore sui ruoli attribuiti ai personaggi femminili in qualsivoglia tipo di prodotto offerto al pubblico: dai contenuti dei loro discorsi, alla qualità e quantità della loro presenza.

1 Data Acquisition

Per la fase di Data Acquisition sono state utilizzate tutte le tecniche presentate a lezione, tra cui l'utilizzo di API, lo scraping di siti Web ed infine il download diretto. Di seguito vengono presentati i dataset considerati all'interno di questo studio.

1.1 Dataset IMDb

Per poter analizzare i film pubblicati negli ultimi 3 anni, si è deciso di utilizzare la piattaforma nota come *Internet Movie Database*, comunemente indicata con l'acronimo IMDb [1]. IMDb è il primo sito cinematografico al mondo, di proprietà di *Amazon*, con un pubblico combinato tra web e dispositivi mobili di oltre 250 milioni di visitatori al mese. Tale sito offre la possibilità di ricercare all'interno di un database più di 185 milioni di informazioni, tra cui più di 4 milioni di film, serie TV e programmi di intrattenimento e più di 6 milioni di membri all'interno di cast e crew.

La piattaforma di IMDb mette a disposizione, pubblicamente e gratuitamente, un'API chiamata *Search Suggestions*. Tale API non è documentata ed è in formato JSON-P pubblico, ciò significa che il dataset scaricato non può essere personalizzato. Pertanto, il dataset ottenuto tramite questa soluzione limita le funzionalità e l'utilità, ad esempio, per eventuali suggerimenti di ricerca che può fornire oppure per esigenze differenti dal dataset fornito, come la lingua o le preferenze di formattazione. Perciò, si è deciso di utilizzare i dataset forniti dal sito IMDbws [2]. Tale sito permette di effettuare il download di sottoinsiemi di dati, presso il sito di IMDb, per uso personale e non commerciale, consentendo di conservare copie locali e soggette ai termini e condizioni di IMDb.

Per questo studio si è deciso di considerare i seguenti dataset:

- `title.basics.tsv.gz`, contenente le seguenti informazioni per i diversi titoli:
  - `tconst` (stringa), identificatore univoco alfanumerico del titolo;
  - `titleType` (stringa), il tipo/formato del titolo (ad es. *film*, *cortometraggi*, *serie tv*, *episodio tv*, *video*, etc);
  - `primaryTitle` (stringa), titolo più popolare/il titolo utilizzato dai registi sui materiali promozionali al momento del rilascio;
  - `originalTitle` (stringa), titolo originale, e pertanto nella lingua originale;
  - `isAdult` (booleano), assume valore "0" se il film non è per adulti e "1" se il film è per adulti;
  - `startYear` (YYYY), rappresenta l'anno di uscita di un titolo (nel caso delle serie TV, è l'anno di inizio della serie);
  - `endYear` (YYYY), in riferimento all'anno di fine Serie TV, assume valore "\N" per tutti gli altri tipi di titolo
  - `runtimeMinutes`, fa riferimento al runtime principale, del prodotto cinematografico considerato, in minuti;
  - `genres` (array di stringhe), include fino a tre generi associati al titolo;

- `title.ratings.tsv.gz`, contenente le seguenti informazioni:
  - `tconst` (stringa), identificatore univoco alfanumerico del titolo;
  - `averageRating`, media ponderata di tutte le valutazioni dei singoli utenti;
  - `numVotes`, numero di voti ricevuti dal titolo.

Ciascun set di dati è contenuto in un file formattato con valori separati da tabulazione (TSV) compresso con gzip nel set di caratteri UTF-8. La prima riga di ogni file contiene intestazioni che descrivono cos'è presente in ogni colonna. Inoltre, il valore "\N" viene utilizzato per indicare che un campo è mancante o nullo per quel titolo (ovvero prodotto cinematografico).

1.2 Dataset Incassi

Si è deciso, inoltre, di utilizzare un ulteriore dataset relativo agli incassi effettuati dai film negli ultimi tre anni. Per tale motivo, si è utilizzato un sito web statunitense di proprietà di Amazon noto come *Box Office Mojo* [3], che raccoglie i dati riguardanti gli incassi cinematografici. Nel sito sono presenti le classifiche dei maggiori incassi in Nord America e a livello mondiale, che vengono aggiornate quotidianamente, con le relative date di uscita di ogni film e i loro rispettivi budget di produzione.

Siccome non è disponibile il download dei dati e neanche la possibilità di utilizzare un'API, per poter ottenere le informazioni necessarie per questo studio, si è deciso di utilizzare le classifiche degli incassi dei film relativi alle annate 2019, 2020, 2021 e di prelevare i dati tramite scraping del sito nella sezione "Worldwide", in cui si mostra la classifica degli incassi dei film a livello internazionale, per anno. Tali classifiche contengono le informazioni del botteghino di 200 film. Facendo clic su un anno specifico si accede alla classifica di un anno, che elenca tutte le versioni disponibili per quell'anno. Per ogni anno si è deciso di acquisire i dati relativi a 5 attributi ovvero:

- `Titles`, ovvero il titolo dei film;
- `Worldwide`, somma degli incassi indicati negli attributi `Domestic` e `Foreign`;
- `Domestic`, si riferisce ai ricavi lordi del botteghino del Nord America (Stati Uniti, Canada e Porto Rico), se non diversamente specificato;
- `Foreign`, includono tutti gli altri paesi non considerati nell'attributo `Domestic`.

Successivamente si è deciso di aggiungere l'attributo `Year`, per diversificare i dati appena acquisiti secondo l'anno di produzione dei film (tale attributo è servito in seguito durante lo studio per verificare la consistenza dei dati in riferimento ai film considerati nel dataset di IMDb).

1.3 Dataset premi Oscar

Per la costruzione di un dataset, che contenesse i vincitori dei premi Oscar nel 2020 e nel 2021 (ovvero in riferimento ai film prodotti nel 2019 e 2020), si è innanzitutto scaricato dalla piattaforma `Kaggle` un dataset già esistente contenente tutti i film premiati o nominati agli Oscar dal 1927 al 2020 [4]. Tale dataset è contenuto all'interno di un file in formato CSV con i seguenti attributi:

- `year_film`, l'anno di produzione del film;
- `year_ceremony`, l'anno della cerimonia;
- `ceremony`, il numero della cerimonia;
- `category`, il nome del premio;
- `name`, che contiene il nome dell'attore, dell'attrice o del regista che ha vinto il premio oppure contiene il valore `NaN` se il premio è stato assegnato al film;
- `film`, che contiene il titolo del film o il valore `NaN` per i premi onorari assegnati alle singole persone;
- `winner`, booleano che assume il valore `True` per i film vincitori e `False` per i film nominati.

Il seguente file è stato caricato in un DataFrame all'interno della piattaforma `Google Colab` e manipolato attraverso il linguaggio Python, usando la libreria `Pandas` [5] e, attraverso l'utilizzo di funzioni tramite questa libreria, si sono rimossi tutti i record di film antecedenti al 2019. Inoltre, sono stati rimossi gli attributi `year_ceremony` e `ceremony`, irrilevanti per lo scopo del progetto.

Per completare questo dataset, con i film premiati o nominati agli Oscar del 2021, si è effettuata una procedura di scraping dal sito ufficiale dell'*Academy of Motion Picture Arts and Sciences* [6], usando la libreria di Python `BeautifulSoup`[5].
All'interno del sito `https://www.oscars.org/oscars/ceremonies/2021` è contenuto l'elenco dei diversi premi assegnati e per ogni premio è presente sia il vincitore che l'elenco dei film candidati per tale premio. Attraverso l'utilizzo della funzione `find_all()` della libreria `BeautifulSoup` si è costruito, con opportuni parametri di ricerca, un elenco di tutti i premi riconosciuti nel range temporale considerato.

Come secondo passaggio si sono divisi tre categorie di premi: quelli in cui il premiato era un singolo attore o attrice, quelli in cui si premiava la miglior canzone e quelli dove il premiato era il film, ricercando le parole *Actor* e *Actress* o *Original Song* all'interno della stringa del nome del premio. Per i primi, sfruttando la funzione `find()` si è isolato il nome dell'attore o attrice premiato e il nome del film, successivamente, con la funzione `append()`, si è inserito il film all'interno del dataframe preesistente. Per il premio alla miglior canzone, invece, si è salvato nella colonna `name` il titolo della canzone e usando le funzioni `strip` e `split` di `Pandas` si è estrapolato il titolo del film da una stringa che conteneva anche informazioni sull'autore della canzone, per poi salvarlo nell'attributo `film`. Per i premi attribuiti ai singoli film si è salvato il titolo nella variabile `film` e, le altre informazioni (il nome del regista, dello sceneggiatore, dei costumisti, ...), se presenti, sono state inserite nella variabile `name`.

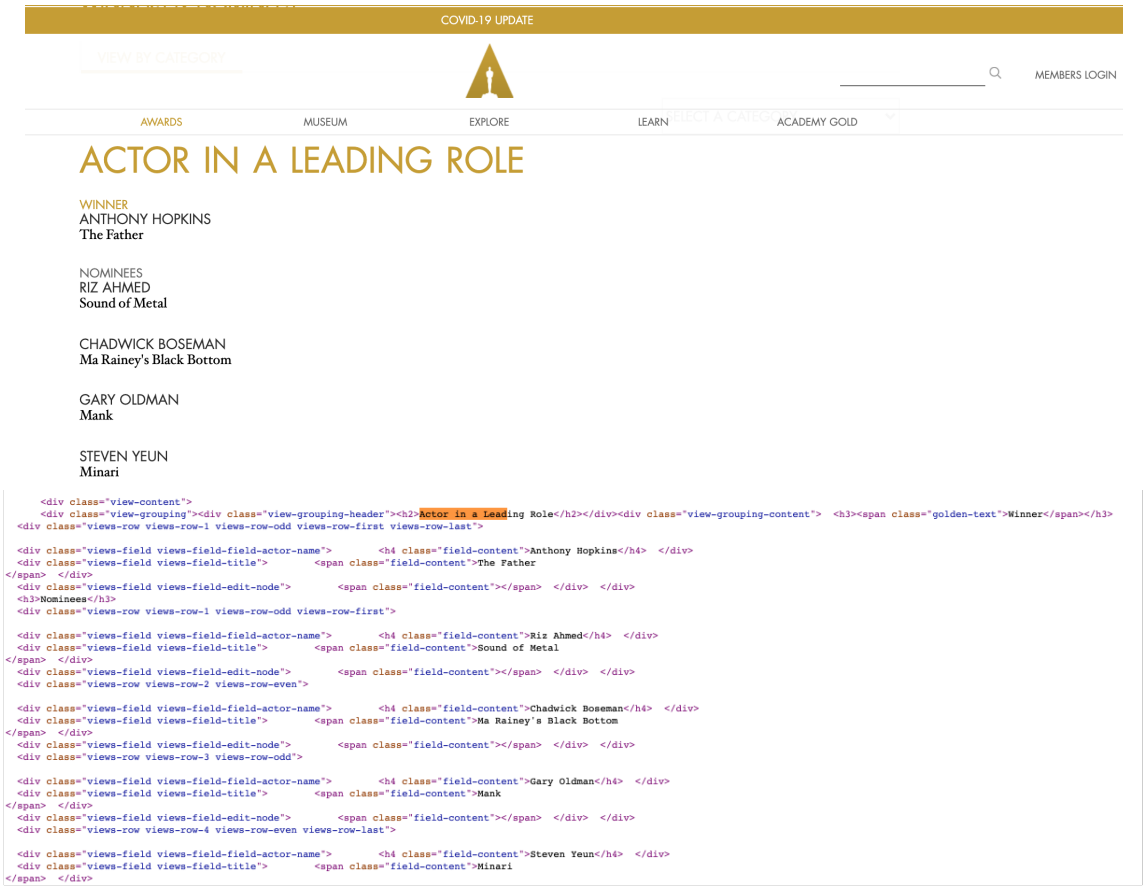


Figura 2: Il sito della *Academy of Motion Picture Arts and Sciences*[6] e il suo file sorgente.

1.4 Dataset Test di Bechdel

Il sito Web *Bechdel Test Movie List* [7] fornisce un'API con 4 metodi principali per acquisire i dati:

1. `getMovieByImdbId`;
2. `getMoviesByTitle`;
3. `getAllMovieIds`;
4. `getAllMovies`.

Le informazioni dettagliate, riguardo questi metodi, si possono trovare nella documentazione API ufficiale del sito Web riportato in precedenza [8].

In primo luogo si è utilizzato il metodo 4, che fornisce un dataset formato da cinque colonne: `title`, `imdbid`, `id`, `rating` e `year`. Il dataset contiene tutti i film archiviati nel sito in ordine cronologico, perciò alla fine del dataset si trovano i dati più recenti. Il dataset è stato salvato all'interno di un dataframe di `Pandas` ed all'interno di un file, in formato CSV, denominato: `Bechdel_originale.csv`.
Tra le informazioni, che il metodo precedentemente usato ci fornisce, l'attributo `imdbid` è l'identificativo dei film utilizzato dal sito di IMDb [1], `id` è l'id univoco del sito Web

utilizzato per scaricare tali dati, `rating` è il punteggio ottenuto nel test di Bechdel dal film ed, infine, `year` è l'anno di uscita del film considerato.

Il punteggio del test, o valutazione, viene calcolato controllando i tre criteri enunciati all'interno dell'introduzione. Ogni criterio è costruito rispetto al precedente, vale a dire che un film non può soddisfare il secondo criterio se il primo criterio non è stato già rispettato. Un punteggio pari a `0` significa che un film non ha due personaggi femminili; `1` significa che un film ha due personaggi femminili, ma non parlano tra loro; pari a `2` esprime che un film ha al suo interno due donne che parlano, ma parlano di qualcosa inerente ad un uomo. Infine, un punteggio pari a `3` indica che il test di Bechdel è stato superato completamente.

Il passaggio successivo è stato eliminare gli anni non facenti parte di questo caso di studio e ordinare l'indice esterno del dataset. Il risultato è stato salvato in un altro file, in formato CSV, nominato: `Bechdel.csv`.

Per avere maggiori informazioni per ogni film, si è utilizzato, inoltre, il metodo 1. Tale metodo utilizza come parametro in entrata l'`imdbid` di un unico filmato e ritorna un oggetto JSON contenente i seguenti attributi del film ricercato: `visible`, `date`, `submitterid`, `rating`, `dubious`, `imdbid`, `id`, `title` e `year`.

`visible` assume sempre il valore `1` per ogni filmato, perché vengono restituiti dal metodo dell'API solo i filmati visibili; `date` è la data di aggiunta del film al sito Web; `submitterid` è l'id di chi ha inserito il dato e infine `dubious` indica se il mittente ha considerato o meno la valutazione dubbia (assume valore `1` nel caso di valutazione dubbia, viceversa assume valore `0`). Siccome le valutazioni di film dubbie non sono affidabili, in quanto suscettibili a modifiche, è possibile scartare tali record, oppure si potrebbe trattare tale dato come un'ulteriore colonna.

All'interno di questo studio si è deciso di tenere tale colonna per un eventuale sviluppo futuro, decidendo quindi di non eliminare i film con valutazione "dubbia". A questo scopo si è iterato il procedimento per ogni film, fornendo al metodo i diversi identificativi `imdbid`, presi dal dataframe salvato in `Bechdel.csv`. Il risultato del ciclo è stato salvato in un dataframe di Pandas con un totale di dodici colonne: `title`, `year`, `rating`, `dubious`, `imdbid`, `id`, `submitterid`, `date`, `visible`, `description`, `status` e `version`. Gli ultimi tre attributi sono generati automaticamente, poiché c'è stato qualche errore durante la creazione del dataset (si veda la sezione relativa alla data preparation del Test di Bechdel), quindi si è proceduto al salvataggio del dataframe come: `Bechdel_detailed_originale.csv`.

2 Data Preparation

In questa fase del progetto sono stati controllati i formati dei vari attributi in comune nei dataset e se vi fossero duplicati tra i record. Inoltre, sono state eliminate le informazioni non utili per questo studio, procedura fatta in parte anche nella fase di Data Acquisition. Tutto ciò è stato fatto anche per mezzo dell'utilizzo di una funzione di `Pandas` di Data Profiling [9], che genera un file HTML con le informazioni sui dataset.

2.1 Dataset IMDb e Dataset Incassi

Prima di procedere con la manipolazione del dataset di IMDb sono state analizzate le informazioni acquisite e si è deciso di eliminare i seguenti attributi poiché non necessari ai fini di questo studio:

- `originalTitle`;
- `endYear`;
- `runtimeMinutes`.

Inoltre, per l'attributo `startYear` sono stati selezionati i record relativi agli anni 2019, 2020 e 2021 e per l'attributo `titleType` si è deciso di selezionare solamente i valori `movie` e `shorts`, eliminando i restanti valori, poiché non oggetto di questo studio:

- `tvEpisode`;
- `tvMiniSeries`;
- `tvMovie`;
- `tvPilot`;
- `tvSeries`;
- `tvShort`;
- `tvSpecial`;
- `video`;
- `videoGame`;

Successivamente, al fine di manipolare il dataset relativo a tutte le informazioni acquisite dal sito di IMDB, si è eseguita l'unione dei dataset `title.ratings.tsv.gz` e `title.basics.tsv.gz` utilizzando la funzione `merge` di `Pandas` [5].

Il dataset relativo agli incassi dei film è stato poi usato come arricchimento dei dati ottenuti tramite IMDb. Per effettuare questo incremento di informazioni, inizialmente, è stata aggiunta la colonna `titleId` al dataset degli incassi, poiché, avendo fatto scraping durante la fase di data acquisition ed avendo a disposizione solamente l'attributo `Titles`, non era possibile effettuare l'unione dei dataset sul tale attributo, poiché scritti in lingue differenti, pertanto si necessitava del codice identificativo (univoco) di IMDb. Per trovare gli id, è stato utilizzato il sito `Cinemagoer` [10], che fornisce gratuitamente un'API utile a tale scopo.

Durante questa fase sono stati riscontrati diversi problemi, poiché alcuni titoli avevano all'interno della stringa `Titles` il termine *Re-release* ovvero "Riedizione" (ad esempio, *Harry Potter and the Sorcerer's Stone 2020 Re-release*). Siccome l'API non riusciva a riconoscere gli id relativi alle riedizioni dei film e verificando manualmente che le riedizioni non hanno un codice identificativo differente, si è deciso di eliminare il termine superfluo e continuare con l'assegnazione degli id. Conclusa tale fase, si è

utilizzata la funzione `merge` di `Pandas` sulla colonna `titleId` al fine di unire i dataset. La scelta dell'attributo su cui effettuare il `merge` è ricaduta sul codice identificavo, poiché è un elemento univoco rispetto al titolo del film e presente in entrambi i set dei dati.

Una volta ottenuto il dataset arricchito, è stato convertito in formato JSON attraverso l'impiego della funzione `to_json` di `Pandas`. Tale scelta è stata effettuata poiché si è verificato che all'interno dell'attributo `genres`, ovvero i generi dei film, vi sono valori multipli. Pertanto attraverso il modello relazionale (quindi in formato CSV) i valori avrebbero continuato ad essere memorizzati in una stringa separati da virgole, invece tramite il formato JSON è possibile memorizzare un array di stringhe con l'elenco dei valori. In questo modo è possibile effettuare delle query più specifiche sui generi dei film. Un'alternativa per risolvere il seguente problema, mantenendo il modello relazionale, sarebbe stata duplicare i records per ogni valore dell'attributo `genres`, in questo modo però sarebbe aumentata inutilmente la dimensione del dataset, generando ulteriore difficoltà nell'accessibilità dei dati.

Infine, durante la conversione in JSON, si è notato che diversi film assumevano il valore `NaN` nelle informazioni relative agli incassi (dovuto al `merge` effettuato sui due dataset iniziali, poiché di differenti dimensioni, siccome gli incassi hanno circa 200 records per ogni anno e i dati relativi a IMDd sono circa 40 mila). Per questo motivo si è deciso di utilizzare una funzione in `Python` che ha permesso di eliminare i valori nulli e mantenere le informazioni degli incassi solo per i film che possiedono tali informazioni. Il tutto è stato poi salvato nel file `imdb.json`.

A seguito dell'utilizzo della funzione di `Pandas` di Data Profiling [9] è stata successivamente riscontrata la presenza di duplicati all'interno dell'attributo `titleId` e all'interno dell'attributo `primaryTitle`. Per entrambi si è verificato manualmente quale fosse il motivo della duplicazione: per il primo attributo si sono eliminati i record che contenevano un errore di consistenza mentre per il secondo attributo si è verificato che non è un errore ma vi sono effettivamente diversi film pubblicati nel range temporale dal 2019 al 2021 con i medesimi titoli. Infine si è verificato che dall'unione del dataset di IMDb e dal dataset degli incassi gli attributi relative a `Year` erano rispettivamente in formato stringa e in formato integer, pertanto erano state memorizzate due differenti colonne per tali valori. Per questo motivo si è deciso di effettuare una trasformazione dell'attributo da integer a stringa e successivamente è stato ripetuto il `merge` dei dataset su due attributi: `titleId` e `Year`.

2.2 Dataset premi Oscar

Per quanto riguarda il dataset relativo ai premi Oscar assegnati negli ultimi due anni, si sono rimossi i record relativi ai premi onorari, ovvero i riconoscimenti attribuiti ad attori, attrici o registi per la loro carriera. Tali valori non sono rilevanti ai fini del progetto, perché non sono associabili ai film prodotti nel 2019 o nel 2020, quindi sono stati rimossi tutti i record con valore `NaN` all'interno del dataset, usando la funzione `dropna` di `Pandas`. Prima di procedere allo scraping, nella fase di Data Acquisition, inoltre, si erano già rimossi degli attributi dal dataset di partenza scaricato da Kaggle, perché valutati irrilevanti e si erano eliminati tutti gli attributi di film antecedenti al 2019. Il dataset è stato poi salvato all'interno del file `Oscar2019-20.csv`.

2.3 Dataset Test di Bechdel

Il dataset contenuto in `Bechdel_detailed_originale.csv` comprende due records interamente nulli (motivo per cui all'interno della fase di data acquisition si ottenevano degli attributi in più). Tali righe corrispondono ai film che possiedono un valore mancante per l'attributo `imdbid` in `Bechdel.csv`. Sistemati i valori di queste due righe, e dopo aver eliminato le colonne superflue per lo scopo di questo lavoro (`description`, `status`, `version`, `visible`, `submitterid`, `id`, `date`), i due dataframe sono stati salvati all'interno di due file: `Bechdel_finale.csv` e `Bechdel_detailed.csv`.

Eseguendo un ulteriore controllo del dataset `Bechdel_detailed.csv`, sono state trovate tre righe duplicate, pertanto si è deciso di eliminare i record in eccesso utilizzando la funzione `drop` di `Pandas` e successivamente si è salvato il tutto in `Bechdel_detailed_finale.csv`.

Inoltre, per quanto riguarda la coerenza tra i vari dataset, è emerso che l'`imdbid` era stato memorizzato in formati diversi all'interno dei dataset, ovvero si è verificato che i valori di `imdbid` erano valori integer privi del prefisso "tt", pertanto sono stati modificati tutti gli identificativi e sono state salvate le modifiche effettuate all'interno del file `Bechdel_dataset.csv`.
È importante evidenziare che si è effettuata la memorizzazione dei dati dopo ogni operazione effettuata poiché, a seguito di approfondimenti in relazione al dataset in questione, si è riscontrato essere l'approccio migliore per evitare una perdita rilevante di informazioni.

Infine, visionando il file HTML per condurre l'attività di Data Profiling [9] relativo a questo dataset, sembrava ci fossero due ulteriori righe duplicate, invece, si è scoperto essere un caso di omonimia di film, prodotti in anni diversi e con trama diversa, ma stesso titolo.

3 Data Integration

Prima di procedere all'effettiva integrazione dei dataset si è verificata la coerenza tra i dati contenuti nei diversi dataset. Partendo da `Oscar2019-20.csv` e da `imbd.json`, usando la funzione `isin`, si è confrontato l'attributo `film` del primo dataset con l'attributo `primaryTitle` del secondo dataset al fine di controllare se i titoli dei film fossero identificati esattamente dalle stesse stringhe. Tale funzione ha riconosciuto 21 valori di film vincitori o nominati agli Oscar privi di corrispondenze nel dataset di IMDb. Procedendo con l'analisi si è riscontrato che sono presenti 4 film in entrambi i dataset, ma aventi i titoli memorizzati sintatticamente in maniera differente: *Once Upon a Time...In Hollywood*, *Star Wars: The Rise of Skywalker*, *Borat Subsequent Moviefilm: Delivery of Prodigious Bribe to American Regime for Make Benefit Once Glorious Nation of Kazakhstan*, *Dcera (Daughter)* e *The Life Ahead (La Vita Davanti a Se)*. Per questi film si è modificato il titolo nella versione contenuta nel dataset Oscar con un nuovo assegnamento. Per i rimanenti 4 film, che hanno destato problemi (ovvero *A Sister*, *Nefta Football Club*, *In the absence* e *Burrow*), si è verificato che effettivamente non sono presenti all'interno del dataset di IMDb, in quanto film premiati agli Oscar

2020, ma prodotti nel 2018.

Come secondo passaggio, sempre considerando il dataset dei premi Oscar, si è effettuato un raggruppamento sui film per poter contare (per ogni film) il numero di Oscar vinti e il numero di nomination ricevute, salvando il risultato ottenuto all'interno di un dataframe di `Pandas` nominato `premi_df`.

Come modello in cui memorizzare il risultato dell'integrazione dei dataset, si è scelto di utilizzare il modello documentale. Inizialmente, si è subito scartata l'opzione del modello a grafo, poiché, per questo studio, non è necessario avere un focus sulle relazioni tra le varie entità, risulta, invece, fondamentale avere la possibilità di poter effettuare delle nidificazioni, non possibili attraverso l'utilizzo dei modelli Column Based e Key-Value. Questi ultimi, per di più, occupano maggiore spazio rispetto al modello documentale, anche se risultano essere meno complessi nell'utilizzo. Sulla base di questa scelta, come DBMS si è deciso di utilizzare ***MongoDB***, poiché disponibile tramite `Google Colab`, in modo tale da avere a disposizione un ambiente altamente collaborativo per i membri del gruppo.

Per poter usare la funzione `merge` di `pandas` per compiere una FULL OUTER JOIN tra i dataset `imdb.json` e `Bechdel_dataset.csv`, si è rinominato l'attributo `imdbid` di `Bechdel_dataset.csv` in `titleId` (per coerenza con l'altro dataset) e si sono rimossi gli attributi `title` e `year`, poiché già presenti all'interno del dataset di IMDb e compiendo un FULL OUTER JOIN si sarebbero ottenute delle colonne duplicate contenenti le medesime informazioni. A questo punto si è effettuato il FULL OUTER JOIN e si è memorizzato il risultato all'interno del dataframe `df_integrated_pz`.

Studiando il dataframe `df_integrated_pz` si sono trovati 40 film presenti nel dataframe `Bechdel_dataset.csv`, ma assenti in `imdb.json` (e dunque privi di tutti gli attributi presenti nel dataset di IMDb), si è provveduto quindi all'eliminazione di tali record usando la funzione `dropna` di `Pandas` [5] con opportuni parametri.
A questo punto si è effettuato anche il FULL OUTER JOIN tra il dataset parzialmente integrato `df_integrated_pz` col dataframe `premi_df` sugli attributi `PrimaryTitle` e `Year`. È stato scelto di usare entrambi gli attributi, e non solamente il nome del film, perché potrebbero esistere diversi film con lo stesso titolo, ma diverso anno di produzione, in questo modo si cerca di identificare correttamente solo il film effettivamente candidato o vincitore di un premio Oscar. Il risultato della seconda chiamata alla funzione `merge` è stato memorizzato nel dataframe `df_integrated`. Pertanto, il nuovo dataset documentale, risultato dell'integrazione, presenta i seguenti attributi:

- `IMDbID`, ID del film come segnato sul sito IMDb;
- `Title`, titolo del film, coincidente su tutti e tre i dataset da integrare;
- `Type`, specifica il formato dei film (film o cortometraggio);
- `isAdult`, specifica se il film è per adulti (variabile booleana);
- `Year`, anno di produzione del film;
- `Genres`, genere del film;

- `Rating`, media voti del film su IMDb;
- `numVotes`, numero di voti ricevuti dal film su IMDb;
- `Worldwide`, incassi ottenuti dal film in tutto il mondo;
- `Domestic`, incassi ottenuti dal film in Nord America;
- `Foreign`, incassi ottenuti dal film in tutto il mondo a parte il Nord America;
- `OscarNominations`, numero di nomination agli Oscar ricevuti dal film;
- `OscarWin`, numero di Oscar vinti;
- `BechdelScore`, punteggio del film al test di Bechdel, per i film di cui è stato valutato.

Come ultimo passaggio si è caricato il dataframe sul DBMS MongoDB e si sono effettuate alcune query per testarlo. Ad esempio, si sono cercati i film che avessero ricevuto tra le 5 e le 10 nominations agli Oscar e si è ottenuto il seguente risultato:

```
[60] results = movies.find({'OscarNominations' : { '$gt' :  5, '$lt' : 10}})

[61] for result in results:
        print(result)

     'titleId': 'tt10618286', 'titleType': 'movie', 'primaryTitle': 'Mank', 'isAdult': 0, 'Year': 2020,
     'titleId': 'tt1070874', 'titleType': 'movie', 'primaryTitle': 'The Trial of the Chicago 7', 'isAdu
     'titleId': 'tt7131622', 'titleType': 'movie', 'primaryTitle': 'Once Upon a Time... In Hollywood',
     'titleId': 'tt7286456', 'titleType': 'movie', 'primaryTitle': 'Joker', 'isAdult': 0, 'Year': 2019,
     'titleId': 'tt8579674', 'titleType': 'movie', 'primaryTitle': '1917', 'isAdult': 0, 'Year': 2019,
     'titleId': 'tt10618286', 'titleType': 'movie', 'primaryTitle': 'Mank', 'isAdult': 0, 'Year': 2020,
     'titleId': 'tt1070874', 'titleType': 'movie', 'primaryTitle': 'The Trial of the Chicago 7', 'isAdu
     'titleId': 'tt7131622', 'titleType': 'movie', 'primaryTitle': 'Once Upon a Time... In Hollywood',
     'titleId': 'tt7286456', 'titleType': 'movie', 'primaryTitle': 'Joker', 'isAdult': 0, 'Year': 2019,
     'titleId': 'tt8579674', 'titleType': 'movie', 'primaryTitle': '1917', 'isAdult': 0, 'Year': 2019,
```

Figura 3

4 Data Quality

Nella fase di Data Quality le dimensioni di qualità che si è deciso di analizzare, poiché le più coerenti con lo sviluppo di questo progetto, sono: la **completezza** e la **consistenza**. In questa fase della progettazione si è ulteriormente utilizzata la funzione di `Pandas` di Data Profiling, analizzando la qualità dei dati presenti nel dataset finale, ovvero il risultato dell'integrazione di tutti i dataset precedentemente esposti.

4.1 La completezza

Per quanto riguarda la dimensione della completezza non si sono riscontrati particolari problemi, solamente per l'attributo `genres` non è stato possibile eseguire la funzione di Data Profiling, poiché ha come valori degli array, di dimensione troppo estesa e con i valori nulli indicati dalla stringa "\N", pertanto, non riconosciuta come missing value da tale funzione. Per questo motivo si è deciso di eseguire la procedura di Data Quality manualmente attraverso l'utilizzo di funzioni in linguaggio Python. In questo modo si è riscontrata la presenza di circa 250 missing value. Per cercare di risolvere questa mancanza di dati, si è provato ad eseguire una fase di Data Improvement, utilizzando

i dati all'interno del dataset Kaggle [11] e l'API della libreria `imdbpy` [10] per trovare gli identificatori univoci (usata in precedenza durante l'acquisizione e preparazione del dataset di IMDb). Questa operazione non ha avuto successo, poiché non si sono trovati i generi mancanti del dataset, pertanto non verrà presentato il codice di questo tentativo, poiché non si è riscontrata nessuna utilità per il miglioramento del dataset.

Come possibile lavoro futuro si potrebbe verificare se tramite un approccio di scraping del sito Wikipedia si è in grado di ottenere le informazioni desiderate, oppure, l'ultimo approccio possibile sarebbe inserire i dati manualmente (il problema è che si trattano però di circa 250 records con generi mancanti).

Dal file html, generato con la funzione, citata in precedenza, inoltre, si è verificata un'elevata presenza di dati nulli per le variabili relative agli incassi, ai premi vinti ed al test di Bechdel, ma è uno tra i motivi alla base della scelta di utilizzare un modello documentale, ovvero poter memorizzare tali valori solamente per i film a cui sono riferiti.

4.2 La consistenza

Per quanto concerne la dimensione di analisi di consistenza si era già presentato il problema durante la fase di Data Preparation, ad esempio, all'interno del dataset di IMDd. Ovvero, siccome si era riscontrata la presenza di record duplicati per l'attributo `titleId`, si è notato che in realtà presentavano `titleId` uguali, ma `Year` o `primaryTitle` differenti, pertanto si è deciso di eliminare i record errati.

In relazione al dataset ottenuto alla fine della fase d'integrazione, attraverso l'utilizzo della funzione `info()` si è notato che 16 film presenti in `premi_df` non erano contenuti all'interno del dataset composto dall'integrazione dei dati di IMDb e il test di Bechdel. Si può distinguere, in questo sottoinsieme di film, che:

- 4 sono i film di cui, già durante la fase di integrazione, si era notata l'incompletezza con il dataset di IMDb e test di Bechdel;
- 11 sono i film che presentano lo stesso titolo, ma anno di produzione differente, pertanto si è andato a correggere manualmente laddove l'anno era sbagliato all'interno del dataset dei `premi_df`. L'errore è scaturito, perché durante l'acquisizione dei dati si è assunto che l'anno di produzione fosse l'anno precedente alla premiazione considerata, ma in realtà l'arco temporale comprende un range più grande di alcuni mesi;
- infine, ad un solo film è stato cambiato il nome del titolo, perché sintatticamente errato (il titolo *Emma* è stato corretto in *Emma.*), sempre all'interno di `premi_df`.

Dopodiché è stata eseguita di nuovo la procedura di integrazione tra il dataset ottenuto da IMDb e il test di Bechdel con `premi_df` attraverso un'operazione riconducibile al "FULL OUTER JOIN" di SQL sugli attributi `primaryTitle` e `Year` (come precedentemente effettuato).

Successivamente, dei 4 film presenti solo all'interno di `premi_df` (poiché l'anno di produzione è il 2018, ma la premiazione è avvenuta nel 2020) si è deciso di eliminare i record a cui facevano riferimento nel dataset totale (`df_integrated`).

Una volta ricontrollato il dataset totale, si è riscontrato un ulteriore problema: i premi venivano assegnati a film con stesso `primaryTitle` e stesso `Year`, ma `titleId` differenti. Pertanto si è andato manualmente a verificare quale film avesse realmente vinto il premio e successivamente è stato inserito il valore nullo per gli attributi `win` e `nominees` per i film che non hanno ricevuto nessun premio.

Infine è stato ricontrollato il dataset per verificare che non vi fosse altro tipo di errore, inconsistenza o incompletezza. Il risultato è stato salvato in formato JSON, utilizzando, inoltre, la funzione per eliminare i valori `NaN` da tale dataset.

5 Sviluppi futuri

In questa sezione vengono proposti i possibili sviluppi futuri per tale lavoro.

- In primo luogo si potrebbe decidere di analizzare più approfonditamente il dataset relativo al Test di Bechdel e, in particolare, i film con attributo `dubious` diverso da `0`, decidendo anche di eliminarli. Inoltre, si potrebbe approfondire l'argomento aggiungendo informazioni riguardanti la presenza di attrici femminili in un film.

- Una volta avvenuta la cerimonia degli Oscar del 2021 (ovvero nel 2022), si dovrebbero aggiungere le nuove informazioni all'interno del dataset relativo ai premi Oscar.

- Sarebbe interessante ampliare questo lavoro considerando anche la presenza (o viceversa l'assenza) di razzismo nei prodotti cinematografici nei diversi anni, ad esempio aggiungendo un attributo relativo alle nazionalità degli attori presenti nei film e la regione di produzione dei filmati. In questo modo si andrebbe ad ampliare il dominio di studio.

- Si è riscontrato, inoltre, durante la fase di integrazione, che il dataset degli Oscar possiede come valore univoco il titolo del film, pertanto, quando si integrano tali dati con le restanti informazioni, non si ha la completa certezza che l'associazione tra i premi vinti e i film (all'interno del dataset di IMDb) sia corretta, poiché si potrebbe verificare la presenza di omonimia di titoli di prodotti cinematografici all'interno dello stesso anno. Una possibile soluzione potrebbe essere arricchire il dataset degli Oscar aggiungendo l'informazione relativa all'identificativo di IMDb per ogni film.

- Come già anticipato nel paragrafo relativo alla Data Quality, si potrebbe risolvere il problema della mancanza dei valori dei generi di circa 250 film facendo scraping da Wikipedia o, l'ultimo approccio possibile sarebbe inserire i dati manualmente.

Sitografia

[1] *IMDb*. URL: https://www.imdb.com/.

[2] *IMDb datasets*. URL: https://datasets.imdbws.com/.

[3] *Box Office Mojo*. URL: https://www.boxofficemojo.com/?ref_=bo_nb_ydw_mojologo.

[4] *Kaggle Dataset The Oscar Award, 1927 - 2020*. URL: https://www.kaggle.com/unanimad/the-oscar-award.

[5] *Libreria Python Pandas*. URL: https://pandas.pydata.org/docs/.

[6] *Academy of Motion Picture Arts and Sciences*. URL: https://www.oscars.org/.

[7] *Bechdel Test Movie List*. URL: https://bechdeltest.com/.

[8] *API Bechdel*. URL: https://bechdeltest.com/api/v1/doc.

[9] *Pandas Profiling*. URL: https://pandas-profiling.github.io/pandas-profiling/docs/master/index.html.

[10] *Cinemagoer*. URL: https://cinemagoer.github.io/.

[11] *Movies Data*. URL: https://www.kaggle.com/mananjhaveri/imdb-movies-data.

[12] Wikipedia. *Test di Bechdel — Wikipedia, L'enciclopedia libera*. 2021. URL: https://it.wikipedia.org/wiki/Test_di_Bechdel.

[13] *Libreria Python BeautifulSoup*. URL: https://www.crummy.com/software/BeautifulSoup/bs4/doc/.