**DOCUMENTAZIONE PROGETTO**

**INGEGNERIA DELLA CONOSCENZA 2021-2022**

Progetto realizzato in solitaria da **Raffaele Di Anna**, matricola **725811.**

****

**Email istituzionale:** *r.dianna@studenti.uniba.it*

**Link progetto:** https://github.com/Raffaele470/RDsAudioBookManager

**INTRODUZIONE:**

Il progetto da me realizzato è chiamato ***RD’s AudioBook Manager***.

Si tratta di un’applicazione che ha come scopo quello di aiutare l’utente nel prevedere e consigliare il prezzo a cui vendere un AudioLibro e di suggerirne uno in base ai dati forniti in input.

In più è possibile effettuare ulteriori query sulla base di conoscenza.

Ho utilizzato un dataset pre-esistente in rete, contenente le informazioni riguardanti degli AudioLibri, in modo tale da poter effettuare le task preventivate.

Al dataset sono state apportate delle modifiche, aggiungendo alcuni valori, al fine di aver una migliore gestione, come verrà spiegato successivamente.

Inoltre, è stata realizzata una piccola base di conoscenza per effettuare delle interrogazioni su di essa.

**REQUISITI:**

Il progetto è stato realizzato in **Python**, quindi per il suo funzionamento è necessario avere a disposizione un programma che permetta la sua esecuzione. (*ad. Esempio Visual Studio Code*).

Inoltre, sono state importate delle **librerie**, per questo è necessario installarle da terminale:

**scikit-plot** 🡪 usando il comando “***pip install scikit-plot”***

**fsspec** 🡪 usando il comando “***pip install fsspec”***

**sklearn** 🡪 usando il comando “***pip install sklearn”***

**numpy** 🡪 usando il comando ”***pip install numpy”***

**matplotlib** 🡪 usando il comando “***pip install*** ***-U matplotlib”***

**pytholog** 🡪 usando il comando “***pip install pytholog”***

**pandas** 🡪 usando il comando “***pip install pandas”***

Una volta installate queste librerie, sarà possibile avviare l’applicativo correttamente.

**GUIDA**

Per avviare il programma basterà aprire il file “**Starting\_Screen.py”.**

Una volta aperto, sarà visualizzata la **confusion matrix**.

Una volta chiusa quella finestra sarà visualizzato il menù, in cui si avrà la possibilità di selezionare una tra le opzioni presenti.

Per selezionare l’opzione relativa basterà digitare il numero corrispondente e premere invio.

Immagine che contiene testo

Descrizione generata automaticamente

1. **PREVISIONE PREZZO**

Dopo aver effettuato questa selezione, qui si avrà la possibilità di indicare dei dati in input al fine da poter ricevere un suggerimento sul prezzo a cui vendere un audiolibro.

Verrà chiesto di indicare la durata dell’audiolibro da vendere, il rating e il numero di recensioni.

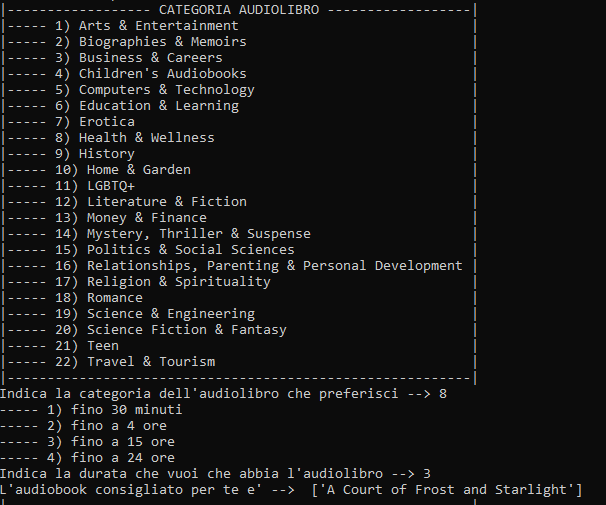
Immagine che contiene testo

Descrizione generata automaticamenteDopo averli inseriti si otterrà il prezzo consigliato.

1. **SUGGERISCI AUDIOLIBRO**

Dopo aver effettuato questa selezione, qui si avrà la possibilità di indicare dei dati in input al fine da poter ricevere un consiglio su quale Audiolibro potrebbe piacere all’utente.

Verrà chiesto di inserire la categoria del libro e la sua durata. Dopo averli inseriti si otterrà l’Audiolibro consigliato.



1. **ACCURATEZZA**

Dopo aver effettuato questa selezione, qui si avrà la possibilità di visualizzare alcune caratteristiche in merito alla **Precision**, **Recall**, **Accuratezza** e **Mae** della previsione.

1. **INTERROGAZIONI SUGLI AUDIOLIBRI**

Dopo aver effettuato questa selezione, qui si avrà la possibilità di effettuare delle interrogazioni per la base di conoscenza sfruttando **Prolog**.

1. **EXIT**

Dopo aver effettuato questa selezione, qui si avrà la possibilità di **uscire** dall’applicazione.

**PROGETTAZIONE APPLICAZIONE**

Il **dataset** utilizzato contiene circa **2000** audiolibri. Originariamente esso era formato da:

*Title, Book Author, Book Narrator, audioRuntime, Categories, average\_rating, ratings\_count e price.*

Per una migliore gestione si sono aggiunte successivamente le informazioni:

*audioRuntime\_converted, average\_rating\_converted, ratings\_count\_converted, category\_converted.*

***audioRuntime\_converted*** *🡪* converte *l’audioRuntime*(la durata) in valori da 1 a 4.

***average\_rating\_converted*** *🡪* converte *l’average\_rating* in valori da 1 a 5.

***ratings\_count\_converted*** *🡪* converte il *ratings\_count* in valori da 1 a 5.

***category\_converted*** *🡪* converte la *categoria* scritta in formato stringa in valori da 1 a 22.

È stato implementato un **DecisionTreeClassifier** per la previsione del prezzo a cui vendere il l’Audiolibro. La decisione del tipo del classificatore è stata fatta confrontando l’accuratezza.

È stata effettuata una comparazione tra il *DecisionTreeClassifier* e il *KNeighborsClassifier*, andando a poi a preferire il primo.

L’accuratezza è stata misurata con la **K-fold Cross Validation**.

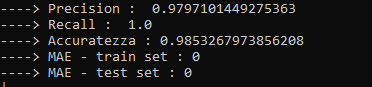
Si allega una tabella contenente i rispettivi dati:

|  |  |  |
| --- | --- | --- |
| **CLASSIFICATORE** | **PRECISION** | **ACCURACY** |
| **DecisionTreeClassifier** | 0.9855072463768116 | 0.99 |
| **KNeighborsClassifier** | 0.9797101449275363 | 0.9853267973856208 |

Da questa tabella si evince la scelta del classificatore per la previsione del prezzo.

Come detto precedentemente è possibile visualizzare le informazioni riguardanti il sistema come:

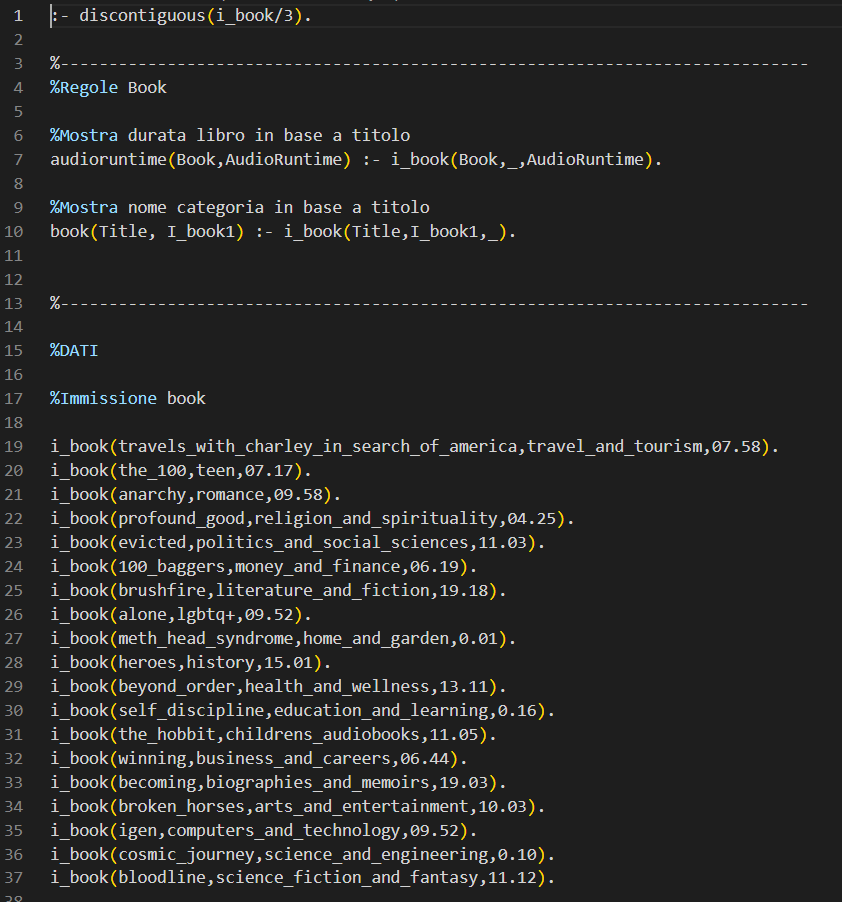
* **Precision & recall** attraverso la funzione nel menù.
* **Mae** così da tener conto dell’errore assoluto tra feature predette e quelle reali che il sistema presenta sia in fase di test che in fase di training.



Per consigliare un AudioLibro è stato implementato invece, un **K-nearest\_neighbors.** In base ai valori ricevuti in input, rappresentanti le caratteristiche dell’Audiolibro da voler trovare, ne verrà consigliato uno dal dataset.

È stato implementato anche **Prolog**, realizzando una piccola base di conoscenza per effettuare delle semplici query. Questa base di conoscenza, come è mostrato anche in foto, è composta da informazioni riguardanti gli Audiolibri, come il loro nome, categoria e durata.

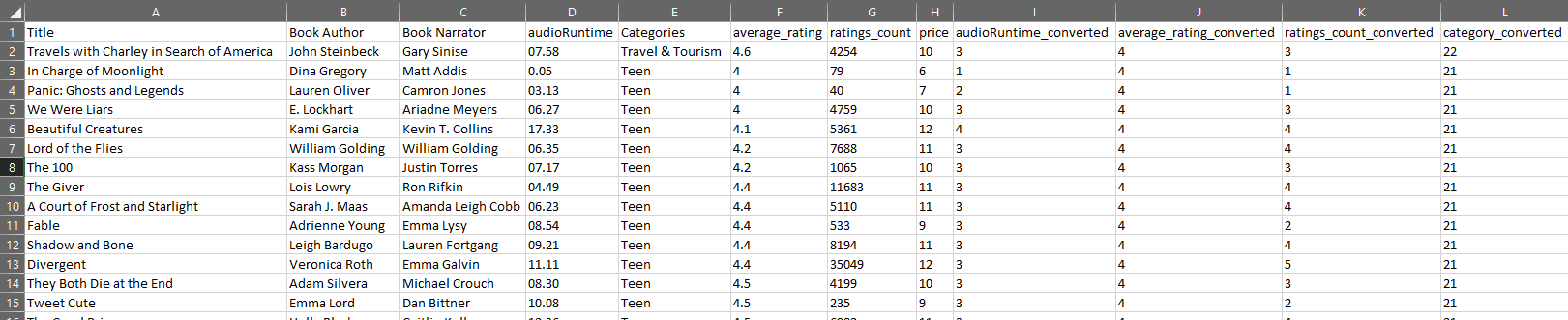
Una volta selezionata la rispettiva voce all’interno del menù iniziale, verrà visualizzato un nuovo menù che consiglierà le possibili query.



Come si può notare abbiamo 2 regole e varie informazioni che possiamo sfruttare per eseguire 4 principali query:

1. **Trovare la durata dell’audiolibro 🡪** inserendo il nome dell’audiolibro sarà possibile trovare la sua durata.
2. **Trovare la categoria dell’audiolibro 🡪** inserendo il nome dell’audiolibro sarà possibile trovare la sua categoria.
3. **Trovare l’audiolibro di una durata specifica 🡪** inserendo la durata sarà possibile trovare l’audiolibro con quella specifica durata.
4. **Trovare l’audiolibro di una categoria specifica 🡪** inserendo una categoria sarà possibile trovare l’audiolibro con quella specifica categoria.

**DATASET**

Il dataset contiene circa 2000 Audiobook, di cui sono presenti diverse informazioni.

Come si può visualizzare dalla foto sono presenti le colonne:

**Title** 🡪 titolo dell’audiolibro

**Book Author** 🡪 autore dell’audiolibro

**Book Narrator** 🡪 narratore dell’audiolibro

**audioRuntime** 🡪 durata dell’audiolibro

**Categories** 🡪 categoria dell’audiolibro

**average\_rating** 🡪 voto medio dell’audiolibro

**ratings\_count** 🡪 numero di voti dell’audiolibro

**price** 🡪 Prezzo dell’audiolibro

**audioRuntime\_converted** 🡪 audioRunTime convertito per motivi progettuali.

**average\_rating\_converted** 🡪 average\_rating convertito per motivi progettuali.

**ratings\_count\_converted** 🡪 ratings\_count convertito per motivi progettuali.

**category\_converted** 🡪 category convertito per motivi progettuali.

**PREVISIONE PREZZO**

Come detto precedentemente, è stata implementata una funzionalità per la previsione del prezzo a cui vendere un Audiolibro dopo aver inserito dei dati di input. Questa funzione sfrutta un **DecisionTreeClassifier** andando ad addestrare grazie al file CSV.

I dati di input chiesti all’utente sono:

* **audioRuntime 🡪** durata dell’audiolibro
* **average\_rating 🡪** voto medio dell’audiolibro
* **ratings\_count 🡪** numero di voti dell’audiolibro

Ma come detto prima, questi sono stati convertiti:

* da **audioRuntime** in **audioRuntime\_converted** sulla base di:

Se *audioRuntime* **<=** **0.30** 🡪 *audioRuntime\_converted* = **1**

Se *audioRuntime* **<= 4** 🡪 *audioRuntime\_converted* = **2**

Se *audioRuntime* **<= 15** 🡪 *audioRuntime\_converted* = **3**

Se *audioRuntime* **> 15** 🡪 *audioRuntime\_converted* = **4**

* da **average\_rating** in **average\_rating\_converted** sulla base di:

se *average\_rating* **>= 5** 🡪 *average\_rating\_converted* = **5**

se *average\_rating* **>= 4** 🡪 *average\_rating\_converted* = **4**

se *average\_rating* **>= 3** 🡪 *average\_rating\_converted* = **3**

se average\_rating **>= 2** 🡪 average\_rating\_converted = **2**

se *average\_rating* **< 2** 🡪 *average\_rating\_converted* = **1**

* **ratings\_count** in **ratings\_count\_converted** sulla base di:

se *ratings\_count* **>= 20000** 🡪 *ratings\_count\_converted* = **5**

se *ratings\_count* **> 5000** 🡪 *ratings\_count\_converted* = **4**

se *ratings\_count* **> 1000** 🡪*ratings\_count\_converted* = **3**

se *ratings\_count* **> 100** 🡪 *ratings\_count\_converted* = **2**

se *ratings\_count* **<= 100** 🡪 *ratings\_count\_converted* = **1**

Inoltre, anche **Categories** è stato convertito in **category\_converted** sulla base di**:**

|  |  |
| --- | --- |
| **Categories** | **Category\_converted** |
| *Arts & Entertainment* | **1** |
| *Biographies & Memoirs* | **2** |
| *Business & Careers* | **3** |
| *Children's Audiobooks* | **4** |
| *Computers & Technology* | **5** |
| *Education & Learning* | **6** |
| *Erotica* | **7** |
| *Health & Wellness* | **8** |
| *History* | **9** |
| *Home & Garden* | **10** |
| *LGBTQ+* | **11** |
| *Literature & Fiction* | **12** |
| *Money & Finance* | **13** |
| *Mystery, Thriller & Suspense* | **14** |
| *Politics & Social Sciences* | **15** |
| *Relationships, Parenting & Personal* | **16** |
| *Religion & Spirituality* | **17** |
| *Romance* | **18** |
| *Science & Engineering* | **19** |
| *Science Fiction & Fantasy* | **20** |
| *Teen* | **21** |
| *Travel & Tourism* | **22** |

Sfruttando questi dati otterremo come risultato il prezzo del libro previsto.



**SUGGERISCI AUDIOLIBRO**

È stata implementata una funzionalità anche per il suggerimento di un libro, sulla base dei valori inseriti in input, sfruttando il **K-Nearest\_Neighbors.**

Attraverso questa funzionalità, verranno chiesti in input alcuni valori:

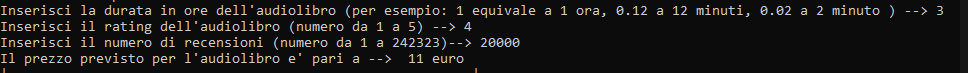
* **category\_converted 🡪** numero indicante la categoria dell’audiolibro
* **audioRuntime\_converted 🡪** numero indicante la durata dell’audiolibro

Immagine che contiene testo

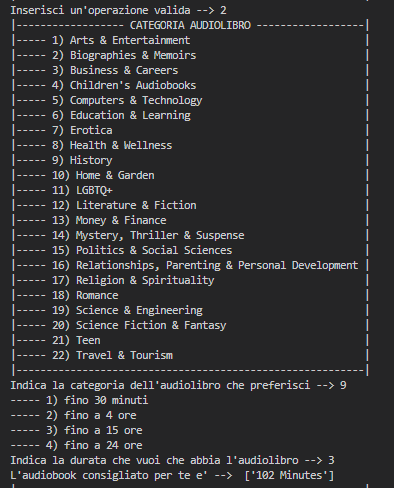
Descrizione generata automaticamenteInfine, quindi, si otterrà un audiolibro consigliato.

**ESEMPI PRATICI**

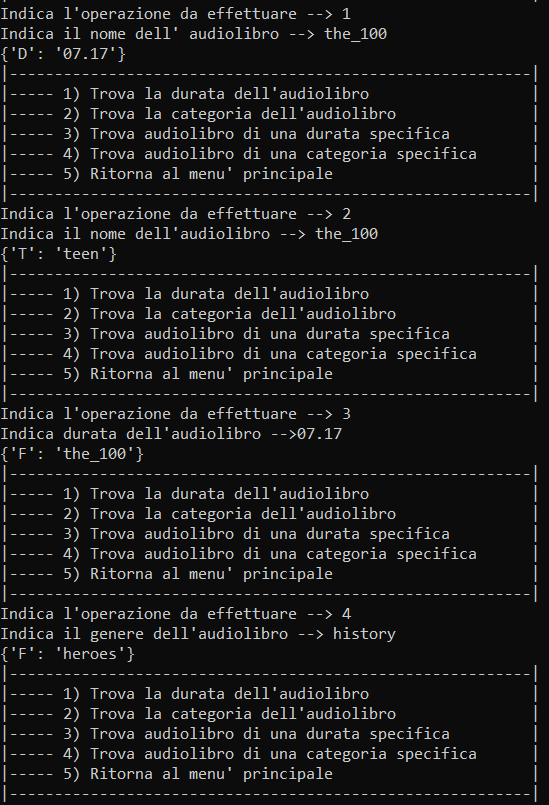
* **Previsione prezzo:**

****

* **Suggerisci Audiolibro:**

****

* **Interrogazioni sugli Audiolibri (prolog):**

****

**CONCLUSIONI**

Il progetto è stato realizzato al fine dello svolgimento dell’esame di **Ingegneria della Conoscenza**, frequentata presso **l’Università di Bari Aldo Moro**.

Personalmente, penso che questo progetto sia stato molto utile per mettere in pratica le conoscenze teoriche acquisite durante il corso e di aver potuto interagire per la prima volta con l’ambiente Python, nonostante le difficoltà riscontrate durante lo sviluppo, soprattutto ritrovandomi a svolgerlo in solitaria.

*Raffaele Di Anna, 725811*