ARTICLE TITLE

FABRIZIO COMINETTI, DAVIDE ABETE, RUBEN AGAZZI

CONTENTS

ABSTRACT

Il progetto realizzato consiste nella creazione di una base di dati a grafo contenente le relazioni fra i vari prodotti Marvel. All'interno del database sono presenti relazioni come ad esempio fra personaggi e fumetti, personaggi e personaggi, etc...

1 INTRODUZIONE

Lo scopo del progetto consiste nella realizzazione di un database a grafo riguardante vari prodotti Marvel, fra cui personaggi, film, serie tv e fumetti. L'obiettivo del progetto è di ottenere una sorta di "rete sociale" di super eroi o personaggi marvel, i quali sono collegati ai rispettivi fumetti, film , serie tv, o rispettivi collegamenti interpersonali fra personaggi e personaggi.

2 FONTI DATI

Per l'ottenimento dei dati sono stati utilizzati due metodi diversi: l'utilizzo di una Web API e Web Scraping.

2.1 Web Scraping

Per la parte riguardante il Web Scraping come sorgente dati è stata utilizzata la Marvel Cinematic Universe Wiki.

Lo scraping è stato effettuato eseguendo un notebook python. I passi eseguiti dal notebook per completare lo scraping sono:

1. Ottenimento dalla pagina relativa a tutti i personaggi della wiki i nomi dei personaggi con i relativi link alla pagina personale.
2. per ogni personaggio aprire la pagina personale e ottenere sempre tramite scraping le informazioni rilevanti, come ad esempio: biografia, lista di film in cui è presente il personaggio, lista di serie tv in cui è presente il personaggio e relazioni con altri personaggi.
3. Per ogni film trovato nelle pagine dei personaggi aprire la pagina relativa al film e ottenere tramite scraping le informazioni del film come ad esempio la trama, i registi, scrittori, compositori, incasso, data di uscita e durata del film.
4. Per ogni serie trovata nelle pagine dei personaggi aprire la pagina relativa alla serie e ottenere tramite scraping le informazioni della serie in questione, come ad esempio la trama, i registi, i produttori e i compositori
5. Salvataggio temporaneo all'interno di file csv di film, serie tv e personaggi per essere processati in seguito.

2.2 Web Api

Per l'ottenimento dei dati tramite web API è stata utilizzato il servizio formito dalla Marvel che mette a disposizione una sua Web Api per ottenere i dati relativi a personaggi e fumetti. L'ottenimento dei dati è avvenuto tramite esecuzione di un notebook python. L'esecuzione consiste nei seguenti passi:

1. Ottenimento dei dati dei personaggi in formato JSON tramite chiamata al relativo endpoint della web api a gruppi di 100 personaggi, in quanto è il limite imposto dagli sviluppatori del servizio.
2. Salvataggio su file CSV dei dati relativi ai personaggi ottenuti.
3. Ottenimento dei dati dei fumetti in formato JSON tramite chiamata al relativo endpoint della web api a gruppi di 100 fumetti, in quanto è il limite imposto dagli sviluppatori del servizio.
4. Salvataggio su file CSV dei dati relativi ai personaggi ottenuti.

3 DATA EXPLORATION

La fase di data exploration è stata realizzata sempre utilizzando python con alcune librerie grafiche per la creazione di visualizzazioni.

3.1 Data Quality

Le principali problematiche riscontrate nella qualità dei dati sono state riscontrate nelle biografie dei personaggi e nelle relazioni tra personaggi ottenute tramite web scraping;

3.1.1 Data Quality e Cleaning Biografia

Per quanto riguarda la qualità dei dati delle biografie le principali problematiche sono:

1. Il notebook salva i vari paragrafi delle biografie come una lista di stringhe, quindi prima di tutto la biografia viene trasformata in una stringa, concatenando le varie stringhe presenti nella lista e inserendo un carattere newLine fra un elemento e l'altro.
2. Il notebook inoltre salvava alcuni paragrafi della biografia più volte, per questo prima di concatenare la stringa viene effettuato un controllo per vedere se la porzione di stringa è già stata aggiunta alla stringa finale
3. Infine viene fatto un escaping dei caratteri speciali, come ad esempio il carattere " o ', in modo da non avere problemi nell'inserimento nel database dei dati.

3.1.2 Data Quality e Cleaning Relazioni

Per quanto riguarda la qualità dei dati delle relazioni interpersonali ottenute tramite web scraping la principale problematica consiste nell'assenza in alcune relazioni del nome del personaggio interessato, ad esempio ci sono relazioni che hanno come nome personaggio "Mother", "Father" e così via. Per risolvere questa problematica il notebook esegue le seguenti operazioni:

1. Per ogni personaggio vengono recuperate le relative relazioni.
2. Per ogni relazione ottenuta viene effettuato un controllo sul nome del personaggio della relazione, se il nome non è presente all'interno della lista di personaggi la relazione viene scartata

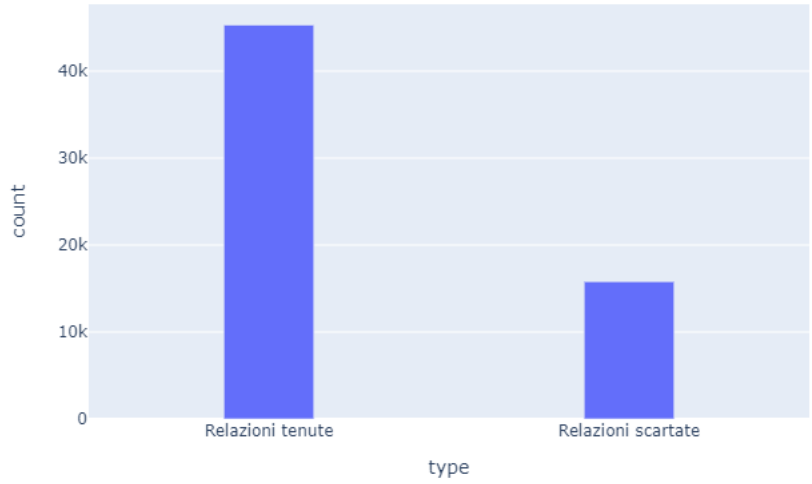

Figure 1: Grafico relativo alle relazioni tenute rispetto a quelle scartate

3.2 Completezza

Sono state effettuate altre analisi riguardanti la completezza dei dati: in particolare al riguardo di attributi dei dati ottenuti.

3.2.1 Dati personaggi

Le analisi di completezza effettuate sui personaggi consistono nel confronto fra i dati ottenuti dalla web api e quelli ottenuti tramite web scraping. In particolare si può riscontrare una corrispondenza di 323 personaggi della web api con quelli ottenuti tramite web scraping, al netto dei 1559 personaggi della web api.

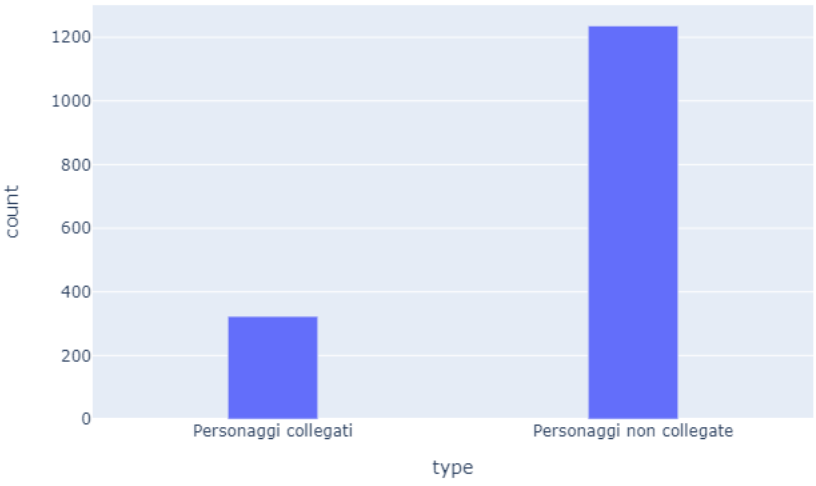

Figure 2: Grafico al numero di personaggi che sono collegabili con i dati ottenuti tramite web scraping.

3.2.2 *Descrizione fumetti*

Le analisi di completezza effettuate sulle descrizioni dei fumetti, ottenute tramite web api, consiste nel vedere quanti fumetti possiedono una descrizione. In particolare circa 19000 fumetti non hanno descrizione al netto dei circa 50000 fumetti.

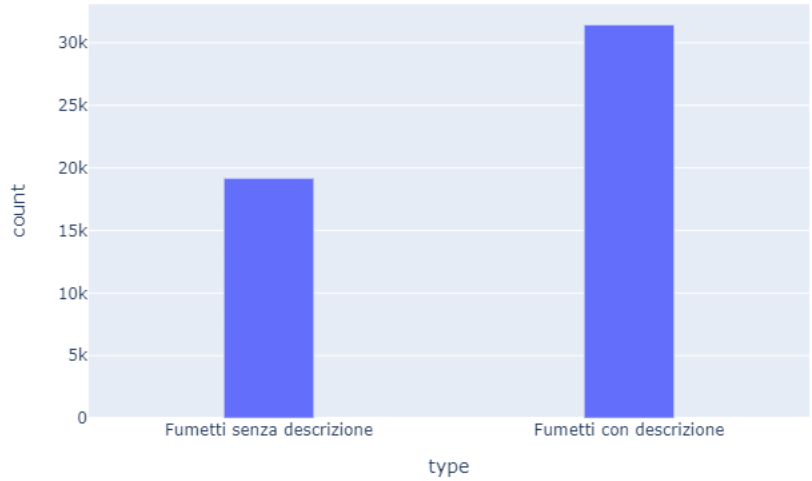

Figure 3: Grafico relativo al numero di fumetti con e senza descrizione

4 DATA INTEGRATION

Per il processo di Data Integration sono state collegate dove possibile le due sorgeti dati tramite gli schemi dei personaggi, i quali fungono da punto di incontro fra le due sorgenti dati

4.1 Schema transformation

Per la fase di schema transformation si aggiunge una nuova colonna al dataset del web scraping e della web api, contenente il nome del personaggio senza alcuni caratteri particolari come ad esempio " o ’ e senza qualsiasi porzione di testo racchiusa fra delle parentesi. Le parentesi sono tolte per poter fare il matching dei personaggi senza tenere conto in questo caso della singola variante del personaggio(ad esempio stesso personaggio di universi differenti). In output si avranno i due schemi relativi ai personaggi con un campo nome processato come detto precedentemente.

4.2 Schema matching

La fase di schema matching avviene durante il processo di inserimento nel database a grafo, utilizzando la seguente regola di integrazione:

1. Per ogni personaggio, sia proveniente da scraping che da web api, creo, se non esiste, un nodo «character» all’interno del database a grafo, utilizzando come discriminante sull’esistenza il nome processato generato nella fase di schema transformation. Se il nodo esiste già non faccio nulla.

2. Per ogni personaggio collego il nodo character corrispondente al nome processato ad un nodo «character variant», che sara creato se non esiste, che è

identificato tramite il nome vero e proprio non processato. Se il nodo esiste già procedo aggiungendo le informazioni aggiuntive relative al personaggio.

4.3 Schemas Integration

Infine, definita la regola di schema matching il notebook procede come segue:

1. Per ogni personaggio della web api eseguo la creazione dei nodi come definito precedentemente
2. Per ogni personaggio ottenuto tramite web scraping eseguo la creazione dei nodi come definito precedentemente
3. Eseguite queste operazioni vengono integrate le informazioni ottenute tramite web scraping alle informazioni già inserite sul database provenienti dalla web api se possibile, sennò vengono creati i nodi corrispondenti ai singoli personaggi in caso di corrispondenza fra i nomi non trovata.

5 DATABASE

5.1 Struttura Database

5.2 Nodi Database

5.3 Relazioni database

5.4 Statistiche database