AMAZON PRIME RECOMMENDATIONS USE CASE STUDY REPORT:

Group No.: Group 25

Student Names: Srinidhi Aduri and Surbhi Virendra Wahie

Introduction:

The Problem:

Users of amazon are increasingly dissatisfied with the recommendations provided by amazon prime video. the data is driven by overall popularity which leads to poor personalization, leading to an increase in customer churn rate.

The Goal:

We aim to solve this problem by using Amazon’s in-house data capabilities. Using Amazon’s rich database, we aim to improve prime user’s movie recommendation system by making use of the user’s Amazon music database.

The Requirement:

Need to generate tags for music that the same user is listing to. This is done by scraping YouTube comments and studying users’ mood that classify contents by genre and microgenre, time-period, plot conclusion, and mood. This tag will help us map and recommend the movies which the user may like.

Conceptual data Modelling

EER Diagram:

Diagram

Description automatically generated

UML Diagram:

Diagram, schematic

Description automatically generated

Mapping Conceptual Model to Relational Model

Relational Model:

USERACCOUNT (USER ID, NAME, EMAILID, PAYMENTID, COUNTRY, PLANCHOSEN,

PAYMENTTYPE)  
EMAILID: FOREIGN KEY FROM LOGINCREDENTIALS – NOT NULL PAYMENTID: FOREIGN KEY FROM PAYMENT – NULL ALLOWED

LOGINCREDENTIALS (EMAILID, PASSWORD)  
PAYMENT (PAYMENTID, PAYMENTMETHOD, AMOUNT, PAYMENTDATE) SUBSCRIPTION (USERID, PAYMENTID, SUBSCRIPTIONTYPE, PRICE,

SUBSCRIPTIONDATE)  
USERID: FOREIGN KEY FROM USERACCOUNT – NOT NULL PAYMENTID – FOREIGN KEY FROM PAYMENT

ACTIVITYHISTORY (USERID, LOGINTIME, LOGOUTTIME, SESSIONTIME,

BROWSETIME, ACTUALTIME) (DO WE NEED A PRIMARY KEY SPECIFICALLY FOR

ACTIVITY HISTORY?)  
USERID: FOREIGN KEY FROM USERACCOUNT  
LIVEVIEWINGHISTORY (SHOWID, USERID, VIEWTIME, RATINGS/LIKES)

GENRE (GENREID, GENRETITLE)  
TVSHOWSANDMOVIES (SHOWID, MOVIETITLE, RELEASEDATE, RATING) SHOWGENRE (GENREID, SHOWID) (DO WE WANT A SHOW TO BE TAGGED TO MULTIPLE GENRES?)  
CAST (SHOWID, ACTORNAME, STAGENAME)

\*ORDERDETAILS (ORDERID, ITEMID, ITEMNAME, ITEMPRICE, ORDERDATE)

ORDERHISTORY (USERID, ORDERID)

ORDERID: FOREIGN KEY FROM ORDERDETAILS

\*MUSICDATA (MUSICID, MUSICNAME, GENRE, ARTIST, ALBUM)

MUSICHISTORY (USERID, MUSICID)

USERID: FOREIGN KEY FROM USERACCOUNT MUSICID: FOREIGN KEY FROM MUSICHISTORY

TAGS (ID, TAG, RELEVANCE)

RECOMMENDATION (USERID, MOVIEID, GENRE) USERID: FOREIGN KEY FROM USERACCOUNT  
MOVIEID: FOREIGN KEY FROM MUSICDATA

Implementation of a relational Model via MySQL:

We have sourced the data from Kaggle. The source file was in csv, excel file format and used SqlAlchemy package to impot data into MySql database using Python.

Text

Description automatically generated

Database Name: Amazon\_Prime

CREATE TABLE `amazon\_prime\_titles` (

`index` bigint DEFAULT NULL,

`show\_id` text,

`type` text,

`title` text,

`director` text,

`cast` text,

`country` text,

`date\_added` text,

`release\_year` double DEFAULT NULL,

`rating` text,

`duration` text,

`listed\_in` text,

`description` text,

KEY `ix\_amazon\_prime\_titles\_index` (`index`)

) ;

Implementation of a relational Model via NoSQL:

Imported a couple csv files into Neo4j and created relations between them.

Graphical user interface

Description automatically generated

Diagram, schematic

Description automatically generatedHere is the magnified version of how our cypher database tables looks like: Diagram, schematic

Description automatically generated

Query for using the tables in the database to get the list of most listened genres.

Graphical user interface, application

Description automatically generated

Database access via Python:

Using the connection code to connect to the MySQL database. Imported SqlAlchemy, created connection, passed the database configurations to create engine, and used the engine to interact with MySQL. Below is the screen capture demonstrating the retrieval of desired data from our Amazon\_Prime database and simple analytics based upon the data.

Graphical user interface, text, application, email

Description automatically generated

Summary and Recommendation

As existing users of Amazon prime video, we have developed an understanding of the level of personalization the platform needs when compared to other major players in the streaming services. Through our analysis, we found a way to recommend to customers and identify niche patterns in user behavior trends using Amazon’s in-house cross-domain datasets, making the process **cost-effective.**

By improving recommendations based on persona, we ensure that the customer’s watch time is increased and browse time is decreased, leading to better engagement and reduction in churn rate