INFO 6210 Data Management and Database Design Physical Data Model and Social Media Assignment 2 Report

ABSTRACT

This project aims to find the details of the **Books** through various social media sources like Twitter, and Reddit and other sources like goodreads website, goodreads API call and datasets. This domain is converted into entities that represent consumers, producers and companies. We can make a conceptual and physical database schema of the data. It Is then populated into sqlite database to query the it and gain information by answering questions.

PART I – CONCEPTUAL MODEL

The domain exhibited here is a Books Domain in which the consumer is our User, Producer is Author and Company is Publishers. From the Twitter API, we have gathered Author and Publisher details (like Followers, Favourited etc) along with the Tweets of Author and Publishers which are then stored in the database. Similarly from Reddit API, we have collected Author and User Posts are stored into separate tables. Below are the attributes and tables used along with their keys.

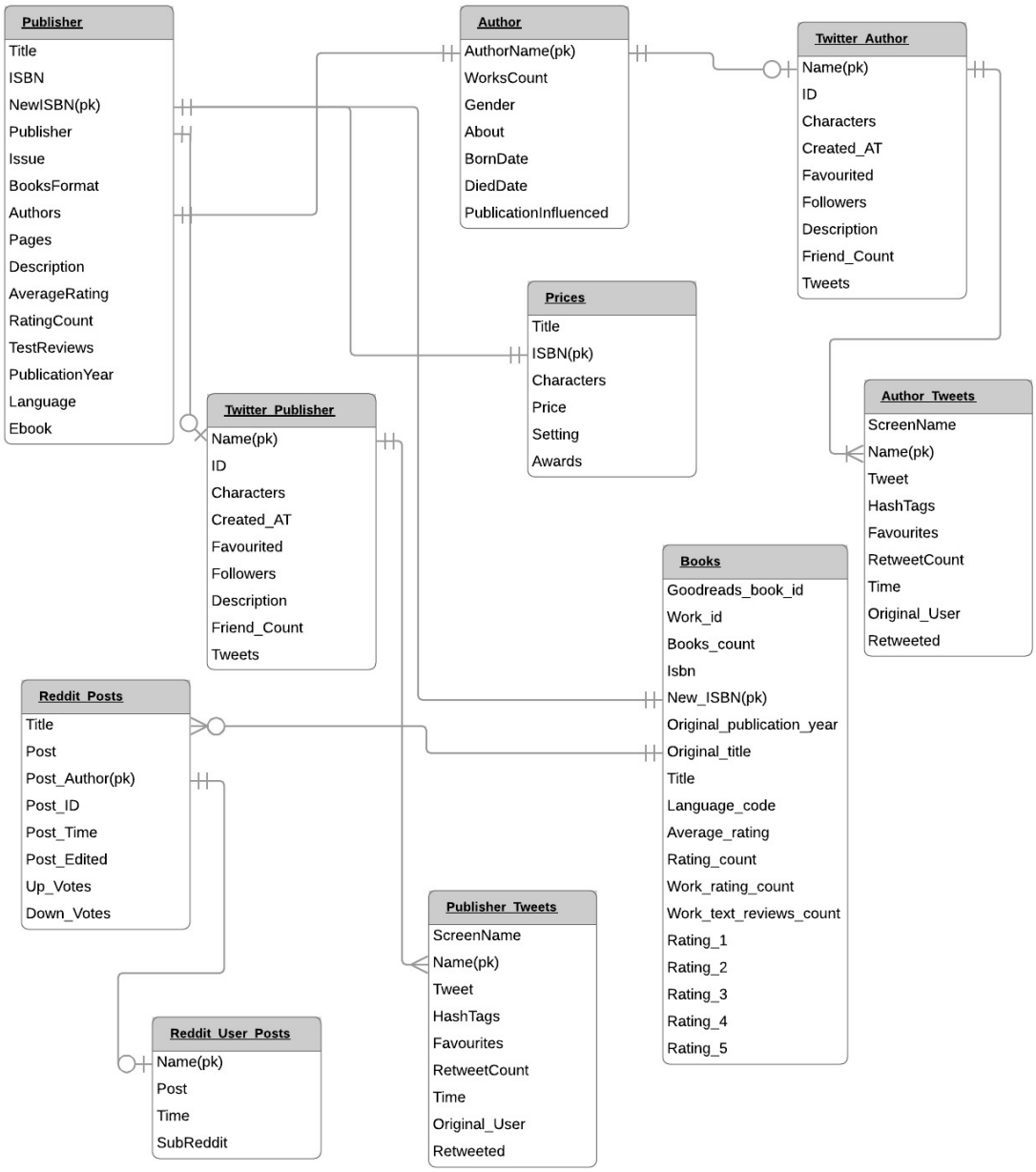
CONCEPTUAL DATABASE SCHEMA

For the company, Nerd Analytics we have designed the conceptual database in such a way that the data is present in the entities accurately and is stored in the sqlite database. Two social media account (Twitter and Reddit) are used in order to get the data and Goodreads website is scrapped for the books data. Other relevant book data is taken from Kaggle data source.

The data is planned to be stored in different relevant tables. From Reddit API, we have created two tables, *Reddit_Posts* and *Reddit_User_Posts* in which we have taken likes, dislikes, posts and hashtags. From the Twitter API, we have gathered Publisher's and Author's followers, tweets, hashtags, retweet counts, friends count and timestamp(when it was posted).This data is distributed accordingly to four tables, *Twitter_Publisher*, *Twitter_Author*, *Publisher_Tweets* and *Author_Tweets.* The other four tables, *Publisher*, *Author*, *Prices* and *Books* are created in a manner which has publisher, author, books and price related detailed information. Now if we want any relevant information we can use joins between these tables and have the desired output.

Below is the conceptual design of the Books domain. Each entity is shown in the box with the Table Name at the top. The Attributes are written below the table name. Each entity is connected with other entity with the cardinalities as depicted by the crow's foot notation. For example, Publisher and Publisher_Tweets has one to many cardinality.

CONCEPTUAL MODEL



QUESTIONS ABOUT CONCEPTUAL MODEL

1) What are the ranges, data types and format of all of the attributes in your entities?

Twitter_Author:- This table contain the Author's description and help us to find his Twitter followers, his tweets etc.

Attribute Name	**Type**
Name	nvarchar(100)
ID	nvarchar(100)
Created_AT	timestamp
Favourited	int
Followers	int
Description	nvarchar(4000)
Friend_Count	int
Tweets	int

Twitter_Publisher:- This table contain the Publisher's description and help us to find his Twitter followers, his tweets etc.

Attribute Name	**Type**
Name	nvarchar(100)
ID	nvarchar(100)
Created_AT	timestamp
Favourited	int
Followers	int
Description	nvarchar(4000)
Friend_Count	int
Tweets	int

Author_Tweets:- This table contain the Author's tweets and help us to know how many people has retweeted his posts, what are his popular hashtags and people who have favourited it.

Attribute Name	**Type**
ScreenName	nvarchar(100)
Name	nvarchar(100)
Tweets	nvarchar(4000)
HashTags	nvarchar(100)
Favourites	int
RetweetCount	int
Time	timestamp
Original_User	nvarchar(100)
Retweeted	nvarchar(100)

Publisher_Tweets:- This table contain the Publisher's tweets and help us to know how many people has retweeted his posts, what are his popular hashtags and people who have favourited it.

Attribute Name	Type
ScreenName	nvarchar(100)
Name	nvarchar(100)
Tweets	nvarchar(4000)
HashTags	nvarchar(100)
Favourites	int
RetweetCount	int
Time	timestamp
Original_User	nvarchar(100)
Retweeted	nvarchar(100)

Reddit_Posts:- This table has the posts and their up votes(like) and down votes(dislike) for different users.

Attribute Name	Type
Title	nvarchar(100)
Post	nvarchar(4000)
Post_Author	nvarchar(100)
Post_ID	nvarchar(100)
Post_Time	timestamp
Post_Edited	nvarchar(100)
Up_Votes	int
Down_Votes	int

Reddit_User_Posts:- This table contains the posts and their popular hashtags(subreddits).

Attribute Name	Type
Name	nvarchar(100)
Post	nvarchar(4000)
Time	timestamp
SubReddit	nvarchar(100)

Author:- This table contains the author details like name, gender, birth date etc.

Attribute Name	Type
AuthorName	nvarchar(100)
WorksCount	int
Gender	nvarchar(100)
About	nvarchar(100)
BornDate	nvarchar(100)
DiedDate	nvarchar(100)
PublicationInfluenced	nvarchar(100)

Books:- This tale has book details and their respective ratings along with publication year.

Attribute Name	Type
Goodreads_book_id	int
Work_id	int
Books_count	int
Isbn	nvarchar(100)
New_ISBN	nvarchar(100)
Original_publication_year	nvarchar(100)
Original_title	nvarchar(1000)
Language_code	nvarchar(100)
Average_rating	nvarchar(100)
Rating_count	int
Work_rating_count	int
Work_text_reviews_count	int
Ratings_1	int
Ratings_2	int
Ratings_3	int
Ratings_4	int
Ratings_5	int

Prices:- This table contains the price of the books and the characters present in that particular book.

Attribute Name	Type
Title	nvarchar(1000)
ISBN	nvarchar(100)
Characters	nvarchar(2000)
Price	nvarchar(100)
Settings	nvarchar(2000)
Awards	nvarchar(4000)

Publishers:- This table has Publishers information and will help us know the books published by a publisher and their reviews.

Attribute Name	Type
Title	nvarchar(1000)
Issue	nvarchar(100)
BooksFormat	nvarchar(100)
Authors	nvarchar(100)
Pages	nvarchar(100)
Description	nvarchar(4000)
AverageRating	nvarchar(100)
RatingCount	int
TestReviews	int
PublicationYear	nvarchar(100)
Publisher	nvarchar(100)
Language	nvarchar(100)

Ebook	int
ISBN	nvarchar(100)
NewISBN	nvarchar(100)

2) When should you use an entity versus attribute? (Example: address of a person could be modeled as either)

   An entity is an independent object which has several attributes where as an attribute is a characteristic that is associated to an entity. So, if you would like to drill down further into an attribute of an entity, a separate entity for that attribute can be created and now this entity will have its own attributes. For eg, we have an Author entity that contains data for the Authors, Name, gender etc are the authors attributes. Also, we have an attribute Author in the publisher table because the publisher works with the author to publish books. Here, the field author is an attribute.

3) When should you use an entity or relationship, and placement of attributes? (Example: a manager could be modeled as either)

We choose an entity based on the design and the information required. If we need more information regarding an attribute, we create a separate entity for that attribute so that it can have other attributes associated to it.

4) How did you choose your keys? Which are unique?

   This number is universal and is unique for each book. We have ISBN as an attribute in our Books table that acts as primary key here similarly Publishers use the ISBN for distribution purposes as well hence we have the field ISBN in the Publishers table as well. When we join the Books and the Publishers the attribute ISBN in the Books table acts as the primary key and the attribute in the Publishers table acts as a foreign key.

5) Did you model hierarchies using the “ISA” design element? Why or why not?

   The model is based on “Has a” design model. Each entity is associated to another using a “Has a” relationship. For example, An Author manages their twitter account. Hence, an Author “Has a” Twitter account.

6) Were there design alternatives? What are their tradeoffs entity vs. attribute, entity vs. relationship, binary vs. ternary relationships?

   Yes, there were design alternatives that we have presented. The trade offs can be between entity vs attribute and bimary vs ternary relationships. However, in the present model there cannot be any tradeoff between entity vs relationship.

   Entity vs Attribute: We have created an Author entity but it is also an attribute in the Books table as we need more specifications about the author. Hence, have author as an entity.

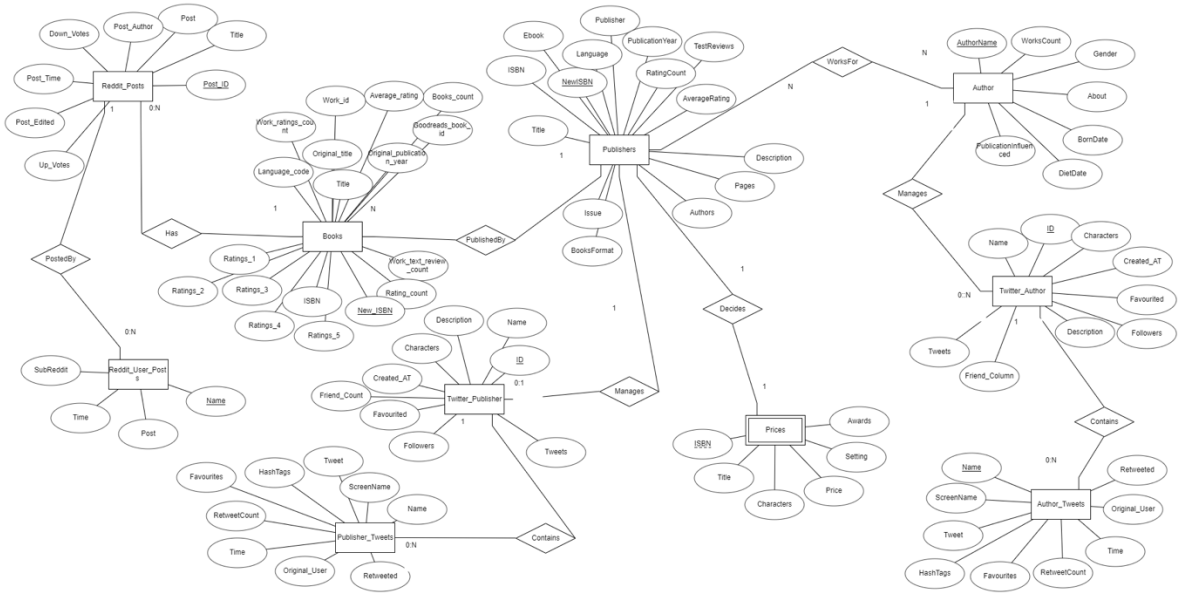
   Binary vs Ternary: We have created a Books table and it is associated with Publisher and Authors. This is identified as a ternary relationship. We have several other binary relationships for eg , Author and their twitter Accounts.

7) Where are you going to find real-world data to populate your model?

Data can be found on Goodreads, Twitter, Kaggle, Reddit.

PART II – PHYSICAL MODEL

ENTITY RELATIONSHIP DIAGRAM



Above is the ER Diagram of the model presented. The entities are represented in the rectangular boxes. Entity price is a weak entity and is shown in double line rectangular box as Price is decided by the Publisher.

The attributes are represented in the circles and each entity has it's own attributes. The primary keys are underlined inside their circles.

The diamond shapes represent the relationship between the entities as how each entity is connected with each other.

QUERIES

1) What user posted this (e.g. tweet, facebook post, IG post, etc.)?

select Post_Author, Post from Reddit_Posts where Post ='Angels & Demons' ;

```
#1.selecting user names who posted about the Angels & Demons Post
pd.read_sql_query("select Post_Author, Post from Reddit_Posts where Post ='Angels & Demons' ;", conn)
```

	Post_Author	Post
0	Hristo21	Angels & Demons

2) When did the user post this (e.g. tweet, facebook post, IG post, etc.)?

select Post_Author, Post_Time, Post from Reddit_Posts where Post ='Angels & Demons' and Post_Author in (select Post_Author from Reddit_Posts where Post ='Angels & Demons') ;

```
#2.selecting time when the user posted about Angels & Demons Post
pd.read_sql_query("select Post_Author, Post_Time, Post from Reddit_Posts where Post ='Angels & Demons' and Post_Author in (selec
```

	Post_Author	Post_Time	Post
0	Hristo21	2019-01-31 16:12:00	Angels & Demons

3) What posts has this user posted in the past 7 days?

select Name, Post, Time from Reddit_User_Posts where (julianday('now') - julianday(Time) < julianday('now') - (julianday('now')-7)) and Name in (select Post_Author from Reddit_Posts where Post ='Angels & Demons');

```
#3.selecting posts that Hristo21 user has posted in last 7 days
pd.read_sql_query("select Name, Post, Time from Reddit_User_Posts where (julianday('now') - julianday(Time) < julianday('now') -
```

	Name	Post	Time
0	Hristo21	Honestly...	2019-03-02 16:48:00
1	Hristo21	We need a right wing	2019-02-27 16:47:00

4) How many post has this user posted in the past 7 days?

select Name, count(Post) as Total_Post from Reddit_User_Posts where (julianday('now') - julianday(Time) < julianday('now') - (julianday('now')-7)) and Name in (select Post_Author from Reddit_Posts where Post ='Angels & Demons') group by Name;

```
#4.selecting count of posts that Hristo21 user has posted in last 7 days
pd.read_sql_query("select Name, count(Post) as Total_Post from Reddit_User_Posts where (julianday('now') - julianday(Time) < jul
```

	Name	Total_Post
0	Hristo21	2

5) What keywords/ hashtags are popular?

select SubReddit_Count,SubReddit from (Select SubReddit, count(SubReddit) as SubReddit_Count from Reddit_User_Posts group by SubReddit) order by SubReddit_Count desc limit 10;

```
#5.selecting top 10 subReddits
pd.read_sql_query("select SubReddit_Count,SubReddit from (Select SubReddit, count(SubReddit) as SubReddit_Count from Reddit_User_
```

	SubReddit_Count	SubReddit
0	557	AskReddit
1	296	ebookdeals
2	274	Fantasy
3	232	politics
4	203	books
5	192	NoSillySuffix
6	183	Showerthoughts
7	175	AutoNewspaper
8	168	funny
9	167	The_Donald

6) What posts are popular?

select Post_Author,Post, Up_Votes from Reddit_Posts order by Up_Votes desc limit 10;

```
#6.selecting top 10 popular posts
pd.read_sql_query("select Post_Author,Post, Up_Votes from Reddit_Posts order by Up_Votes desc limit 10;", conn)
```

	Post_Author	Post	Up_Votes
0	iBlueSweatshirt	Today, NASA will officially have to say goodby...	202549
1	FunnyID	Not just one pick, but two.	129085
2	Mezotronix	Steve Jobs said it first	128609
3	ELFAHBEHT_SOOP	TIL Dennis Ritchie who invented the C programm...	125647
4	ChaseDonovan	TIL that Willie, a parrot, alerted its owner, ...	125492
5	sebaez_	First paralyzed human treated with stem cells ...	124875
6	michaelscottspenis	7 years ago I wanted the cheapest cat ever. Al...	114014
7	ThatDIYCouple	Gabriel Nobre, 19, with his mom and sister rig...	109910
8	LOLerSk8ter	I'm trying to find a guy who lost all his swit...	108530
9	headtailgrep	Canadian school board issues 6000 suspension n...	107232

CONCLUSION

From this Assignment we were able to perform and learned the following:

1. Extract data from Twitter and Reddit APIs.
2. Identify Attributes and entities to create a database.
3. Create associations among several entities.
4. Create a conceptual database.
5. Improve a conceptual database to create a physical database.
6. Write SQL queries.

CONTRIBUTIONS

Rashika Moza: 50%
Akash Srivastava: 50%

CITATIONS

1) https://github.com/nikbearbrown/INFO_6210

2) https://www.sqlite.org/datatype3.html

3) https://www.w3schools.com/sql/default.asp

4) https://praw.readthedocs.io/en/latest/getting_started/quick_start.html

5) http://docs.tweepy.org/en/v3.5.0/

6) https://stackoverflow.com/questions/11131958/what-is-the-maximum-characters-for-the-nvarcharmax

LICENSE