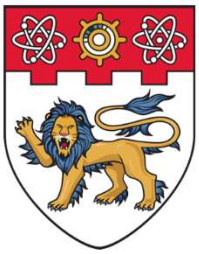


NANYANG TECHNOLOGICAL UNIVERSITY
SINGAPORE

MH8341 Data Management and Business Intelligence

Group 9

Benson Lai Minxian

Li Xiaomeng

Nie Fan

Ong Zhen Ying Melissa

Yong Chai Li

Zhou Xuefei

Contents

About Douban

Douban is a popular social networking site in China which allows registered users to discuss and review books, movies, and music. In 2013, Douban.com had 200 million registered users.[1] The site also proactively recommends books, movies, and music to its users, and even organises offline events.[2]

The traditional revenue streams of Douban.com include income from advertising, interactive marketing, channel fees, and sale of merchandise.[3] More recently in 2020, the company expanded its scope to include publication of original digital works.

For purposes of the project, we will limit the design of the operational database to the review platform for books and movies, and two revenue streams of Douban.com: advertising and e-commerce.

Review platform for books and movies

The review platform for movies is Douban 电影 while the one for books is Douban 读书. Both platforms allow registered users to give ratings and write reviews. A summary of the ratings is presented alongside general information on the reviewed movie or book. An example of the movie and reading site is shown below.

[1] https://en.wikipedia.org/wiki/Douban
[2] https://baike.baidu.com/item/%E8%B1%86%E7%93%A3%E7%BD%91/5549800?fromtitle=%E8%B1%86%E7%93%A3&fromid=7803606
[3] https://www.crunchbase.com/organization/douban

豆瓣读书

书店不死

作者: [日]石桥毅史
出版社: 明室Lucida | 北京联合出版公司
出品方: 明室Lucida
原作名: 「本屋」は死なない
译者: 熊韵
出版年: 2021-10
页数: 312
定价: 55.00元
装帧: 精装
ISBN: 9787559654960

豆瓣评分
7.8 ★★★★☆ 55人评价
5星 25.5%
4星 49.1%
3星 23.6%
2星 1.8%
1星 0.0%

当前版本有售 · · · · · ·

京东商城	55.00元	购买纸质书
当当网 每满100-50	55.00元	购买纸质书
中图网	42.90元	购买纸质书
孔网	25.00元起	购买纸质书

想读 在读 读过 评价: ☆☆☆☆☆
写笔记 写书评 ¥加入购书单 分享到 推荐
+ 加入购书单

书店不死 短评

读过(41) 在读(1) 想读(19) 我来写短评

热门 最新

bird ★★★★☆ 2021-10-23 20:34:33 17 有用

中日两国的书店经营与出版存在着许多差异，但都是为了人与书的相遇。这本书约写于十年前，作者的一些困惑或许已得到解答，如电子书的冲击是否会给书店造成灭顶之灾，某种书店形态到底行不行得通。它依然也能帮我们理解书店的悲情，细数大陆今天我们能看到的书店也不外乎是作者采写的几类，但书中提出的经验总结却仍未被多少书店从业者所认识接纳，开书店生死由命，有时并不考虑怎样更好地活下去，珍惜留住

Advertising revenue

Douban offers display/video ads, content-based ads, and event based/interactive marketing, as shown below.[4] Between 2010 and 2012, Douban.com cooperated with close to 200 brands to provide them with advertising solutions.[5]

豆瓣广告

[4] https://www.douban.com/partner/product#expose-type

[5] https://baike.baidu.com/item/%E8%B1%86%E7%93%A3%E7%BD%91/5549800?fromtitle=%E8%B1%86%E7%93%A3&fromid=7803606

E-commerce revenue

We studied the sales of two products by Douban.

The first is ebooks. The Douban 阅读 platform sells e-books, including digital self-published works. These works can be read on desktop and/or mobile devices. An ebook featured on the website is shown below.

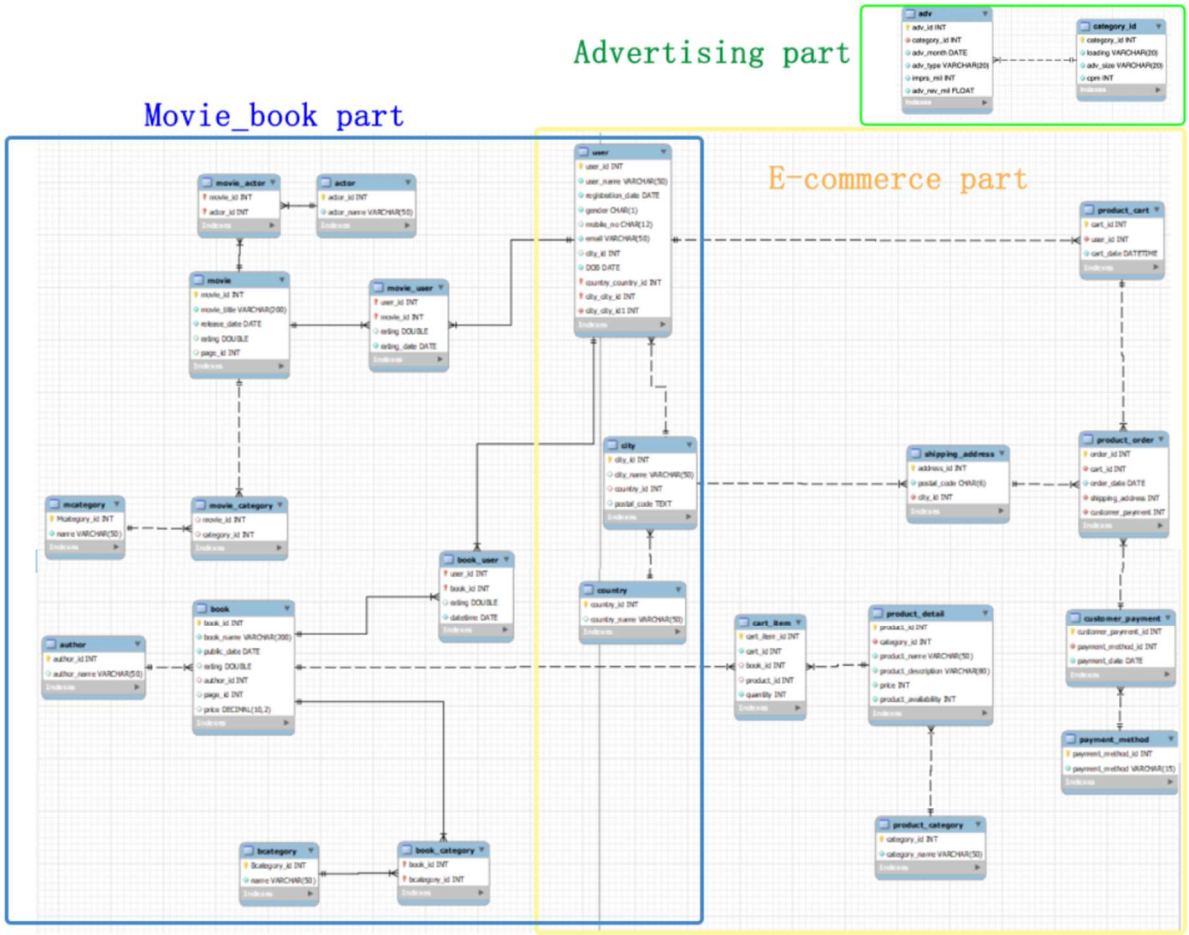
豆瓣阅读

Besides ebooks, Douban also has their own merchandise line named Douban 豆品. It includes calendars, watches, and toys. One of the merchandises, a 2022 calendar, is shown below.

豆瓣豆品

OLTP Database Design

The combined ER diagram is shown below. The breakdown can be seen in the Annex.



OLTP Table Structure

Table name				
Column Name	**Datatype**	**Key**	**Other Constraints**	**Description**
country				
country_id	int	Primary Key (PK)		
country_name	varchar(50)		not null, unique	
city				
city_id	int	PK		
city_name	varchar(50)		not null, unique	
country_id	int	Foreign Key (FK) - country		
user				
user_id	int	PK		
user_name	varchar(20)			
registration_date	date			
gender	char(1)			
mobile_no	varchar(20)			
email	varchar(50)			
city_id	int	FK - city		
DOB	date			
movie				
movie_id	int	PK		
movie_title	varchar(200)			
release_date	date			
rating	double			
page_id	int			
mcategory				
mcategory_id	int	PK		
name	varchar(50)			
movie_category				
movie_id	int	PK	FK – movie	One movie may have one or two categories
category_id	int		FK – mcategory	
actor				
actor_id	int	PK		
actor_name	varchar(50)			
movie_actor				
movie_id	int	PK	FK – movie	A movie can have many actors
actor_id	int		FK – actor	
movie_user				
user_id	int	PK	FK – user	Many users have many ratings to different movies
movie_id	int		FK – movie	
rating	double			
rating_date	date			
author				
author_id	int	PK		
author_name	varchar(50)			
book				
book_id	int	PK		
book_name	varchar(200)			
public_date	date			
rating	double			
author_id	int	FK - author		One book only has one author
page_id	int			

Table name				
Column Name	**Datatype**	**Key**	**Other Constraints**	**Description**
price	decimal			
bcategory				
category_id	int	PK		
name	varchar(50)			
book_category				
book_id	int	PK	FK – book	One book may have one or two categories
bcategory_id	int		FK – bcategory	
book_user				
user_id	int	PK	FK – user	Many users have many ratings to different books
book_id	int		FK – book	
rating	double			
datetime	date			
adv				
adv_id	int	PK		
category_id	int		FK – category_id	
adv_month	date		not null	
adv_type	varchar(20)		not null	
imprs_mil	int		not null	
adv_rev_mil	float		not null	imprs * cpm / 1000
adv_id	int	PK		
category_id				
category_id	int	PK		
loading	varchar(20)		not null	
adv_size	varchar(20)		not null	
cpm	int		not null	
product_category				
category_id	int	PK		
category_name	varchar(50)		not null	
product_detail				
product_id	int	PK		
category_id	int		FK – product_category	
product_name	varchar(50)		not null	
product_description	varchar(80)		not null	
price	int		not null	
product_availability	int		not null	
product_cart				
cart_id	int	PK		
user_id	int		FK – user	
cart_date	datetime		not null	
cart_item				
cart_item_id	int	PK		
cart_id	int		FK – product_cart	Multiple cart items may have the same cart_id
book_id	int		FK – book	
product_id	int		FK – product_detail	
quantity	int		not null	
payment_method				
payment_method_id	int	PK		
payment_method	varchar(15)		not null	
customer_payment				
customer_payment_id	int	PK		

Table name				
Column Name	**Datatype**	**Key**	**Other Constraints**	**Description**
payment_method_id	int		FK – payment_method	
payment_date	date		not null	
shipping_address				
address_id	int	PK		
postal_code	char(6)		not null	
city_id	int		FK – city	
product_order				
order_id	int	PK		
cart_id	int		FK – product_cart	
order_date	date		not null	
shipping_address	int		FK – shipping_adress	
customer_payment	int		FK – customer_payment	

Data generation

We mainly generated data using mockaroo and Excel. Uipath was used to datascrape real websites. For the specific data generation methods for each table within the OLTP diagram, refer to the table below.

Table name	**Method**
movie & book	We generated 20 different categories for books and movies and selected 500 books and 500 movies. Excel was used to set the time users rated each book and movie to be later than the time when book or movie was released, and later than the time when user registered. For the price of the books, UiPath was used to scrape data off the first 500 featured ebooks on Amazon to derive the price range of typical books.
user	Since Douban is a Chinese site, for the location of users, we extracted the list of the most populous cities (top 183 cities) in China and the countries with the highest Chinese population (top 99 countries) from Wikipedia. We used an excel formula to assign a random city to each user based on the proportion of Chinese population in each location. For the demographics of the users, we set the percentage of male users to 40% and female users to 60%. The same excel formula was used to generate the gender. For the age range of the users, we extracted the age distribution of Internet users in China from https://researchgate.net and again, used the excel formula to assign a random date of birth based on proportion.
adv & category_id	Advertising-related data generated is based on Douban's Media Kit, which includes relevant information like advertising costs, loadings, formats, sizes, and so on. Custom advertising like video commercials and full site takeovers are not included for the purpose of data generation, as it is not considered to be de rigueur and evident on Douban website. The real data gleaned from Douban's Media Kit were then referenced to derive and generate some mocked-up data with a view to answer the following analytical questions.

	In addition to the Media Kit, audience information was also referenced to create for semblance of realness and salience that would best answer the analytical questions as realistically as possible.
product_detail	UiPath was used to scrape data from Douban's website, to obtain the product name, description and price of the products listed.
product_cart	The date that the cart was created was generated using Excel formula based on a random date that was after the user's registration date. A total of 49,076 entries were generated.
cart_item	We randomly generated cart items added into the online cart with some carts containing multiple items. We also included some adjustments for seasonal sales as it was observed in Amazon's ebooks sales data that the sale of ebooks would increase in Q3 and Q4 of each year. Since our analysis would be focused on sales of the ebooks, most of our generated sales pertained to ebooks as opposed to other e-commerce products. A total of 50,240 entries were generated.

customer_payment	Sales data generated was limited to the last five full calendar years (i.e., 1 Jan 2016 to 31 Dec 2020) as the data from the past becomes less relevant as time passes.
	
	Based on the online shopping cart abandonment rate from https://statista.com, we used python to randomly select 15,003 carts out of all the carts that would actualise as orders. The python code used is included below:
	
	`import random`
	
	`my_list = list(range(1,6099))`
	`random.shuffle(my_list)`
	
	`# importing pandas as pd`
	`import pandas as pd`
	
	`dict = {'cart_no': my_list}`
	
	`df = pd.DataFrame(dict)`
	
	`# saving the dataframe`
	`df.to_csv('orders.csv')`
	
	We assumed that a higher proportion of users used Alipay and Wechat as payment methods, with only a small percentage using credit card/PayPal. Other data generated like order dates were randomised to a date after the online cart was created.

Analytics

Three business areas were analysed in our project and the resulting OLAP and BI dashboards of our analysis is detailed below.

Review platform for books and movies

Analytical Questions

To gauge Douban's market penetration and cater to the preferences of its users, we anticipate that Douban's management would be interested to know the answers to the following questions:

- How is the user engagement trend in years / months?
- What is the user's distribution by location?
- How does the category of books or movies affect user's rating?
- What is the new increment number of comments (movie & book)?

ETL

Code
1. Location dimension create table location_D as (select city_name, country_name from city left join country on city.country_id = country.country_id); ALTER TABLE location_D ADD location_key INT PRIMARY KEY AUTO_INCREMENT first;
2. Date dimension # import from excel (generate date data from 2010.01.01 to 2021.12.31) alter table date_D modify y_m_d date; alter table date_D add primary key (date_key);
3. Book category dimension create table bcate_middle as(select b.book_id, b.name c1, c.name c2, if(c.name is null,b.name, concat(b.name,' ',c.name)) c from (select * from (select book_id, book_category.bcategory_id, name, rank() over (partition by book_id order by name) r from book_category left join bcategory on book_category.bcategory_id = bcategory.bcategory_id) a where r = 1) b left join (select * from (select book_id, book_category.bcategory_id, name, rank() over (partition by book_id order by name) r from book_category left join bcategory on book_category.bcategory_id = bcategory.bcategory_id) a where r = 2) c on b.book_id = c.book_id); create table bcategory_D as(select c1,c2 from(select distinct c,c1,c2

```
from bcate_middle)a);
ALTER TABLE bcategory_D ADD bcategory_key INT PRIMARY KEY AUTO_INCREMENT first;
```

```
# 4. Movie category dimension
create table mcate_middle as(
select b.movie_id, b.name c1, c.name c2, if(c.name is null,b.name, concat(b.name,' ',c.name)) c
from
(select * from
(select movie_id, movie_category.category_id, name,
rank() over (partition by movie_id order by name) r
from movie_category left join mcategory
on movie_category.category_id = mcategory.Mcategory_id) a
where r = 1) b
left join
(select * from
(select movie_id, movie_category.category_id, name,
rank() over (partition by movie_id order by name) r
from movie_category left join mcategory
on movie_category.category_id = mcategory.Mcategory_id) a
where r = 2) c
on b.movie_id = c.movie_id
);
create table mcategory_D as(
select c1,c2 from(
select distinct c,c1,c2
from mcate_middle)a);
ALTER TABLE mcategory_D ADD mcategory_key INT PRIMARY KEY AUTO_INCREMENT first;
```

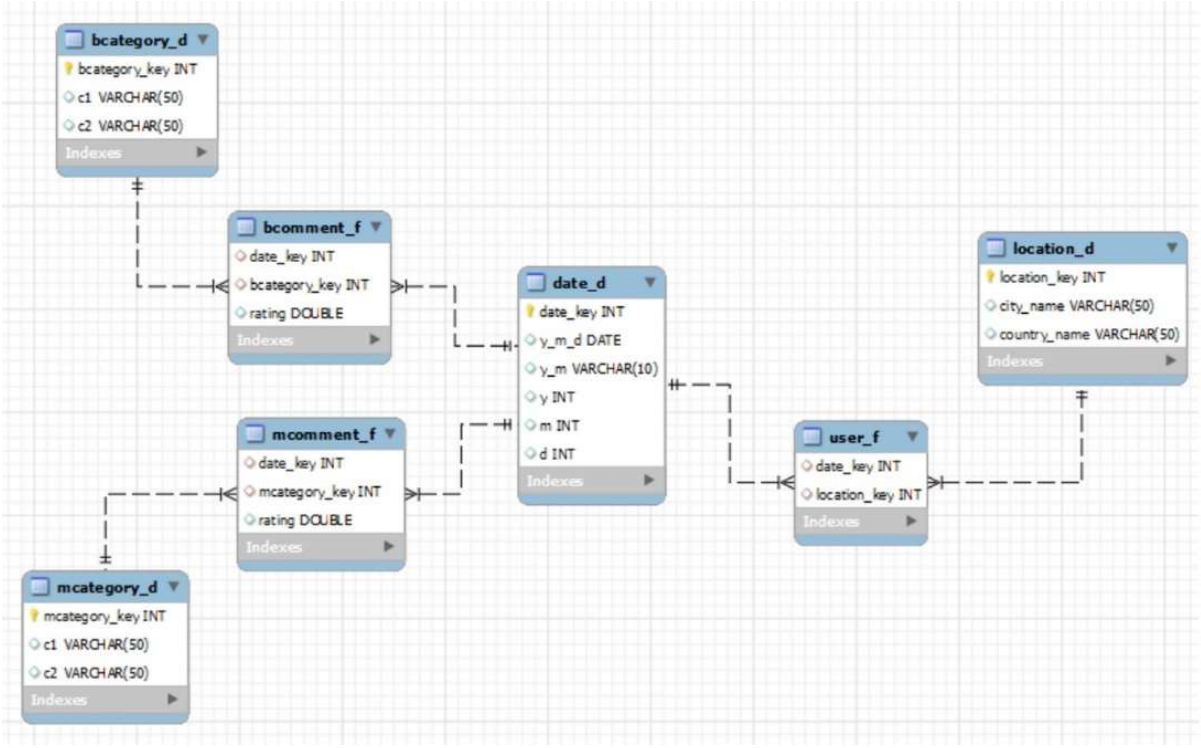
```
# 1. User fact
create table user_F as(
select date_key, location_key
from(
select distinct user_id, user.city_id,date_key, city.city_name, location_key
from user left join date_D
on user.registration_date = date_D.y_m_d
left join city
on user.city_id = city.city_id
left join location_d
on city.city_id = location_d.location_key)a);
alter table user_F add foreign key (date_key) references date_D(date_key);
alter table user_F add foreign key (location_key) references location_D(location_key);
```

```
# 2. Book comment fact
create table bcomment_F as(
select date_key, bcategory_key, rating
from book_user left join date_d
on book_user.datetime = date_d.y_m_d
left join bcate_middle b
on book_user.book_id = b.book_id
left join
(select *, if(c2 is null,c1, concat(c1,' ',c2)) c
from bcategory_d
)sub
on b.c = sub.c);
alter table bcomment_F add foreign key (date_key) references date_D(date_key);
alter table bcomment_F add foreign key (bcategory_key) references bcategory_d(bcategory_key);
```

```
# 3. Movie comment fact
create table mcomment_F as(
select date_key, mcategory_key, rating
from movie_user left join date_d
on movie_user.rating_date = date_d.y_m_d
left join mcate_middle b
on movie_user.movie_id = b.movie_id
left join
(select *, if(c2 is null,c1, concat(c1,' ',c2)) c
from mcategory_d
)sub
on b.c = sub.c);
alter table mcomment_F add foreign key (date_key) references date_D(date_key);
alter table mcomment_F add foreign key (mcategory_key) references mcategory_d(mcategory_key);
```

OLAP Data Warehouse Design

The ER diagram for the Review platform for books and movies is shown below.



OLAP Table Structure

Dimension Table			
Column Name	**Data Type**	**Key**	**Data Source (Table Name)**
date_d			
date_key	int (auto_increment)	surrogate key	
year	int		
month	int		
day	int		
Y_m_d	date		

Dimension Table			
Column Name	**Data Type**	**Key**	**Data Source (Table Name)**
location_d			
location_key	int	surrogate key	
city_name	varchar(50)		city
country_name	varchar(50)		country
bcategory_d			
bcategory_id	int	surrogate key	
c1	varchar(50)	(Category type 1)	bcategory
c2	varchar(50)	(Category type 2)	bcategory
mcategory_d			
mcategory_id	int	surrogate key	
c1	varchar(50)		mcategory
c2	varchar(50)		mcategory

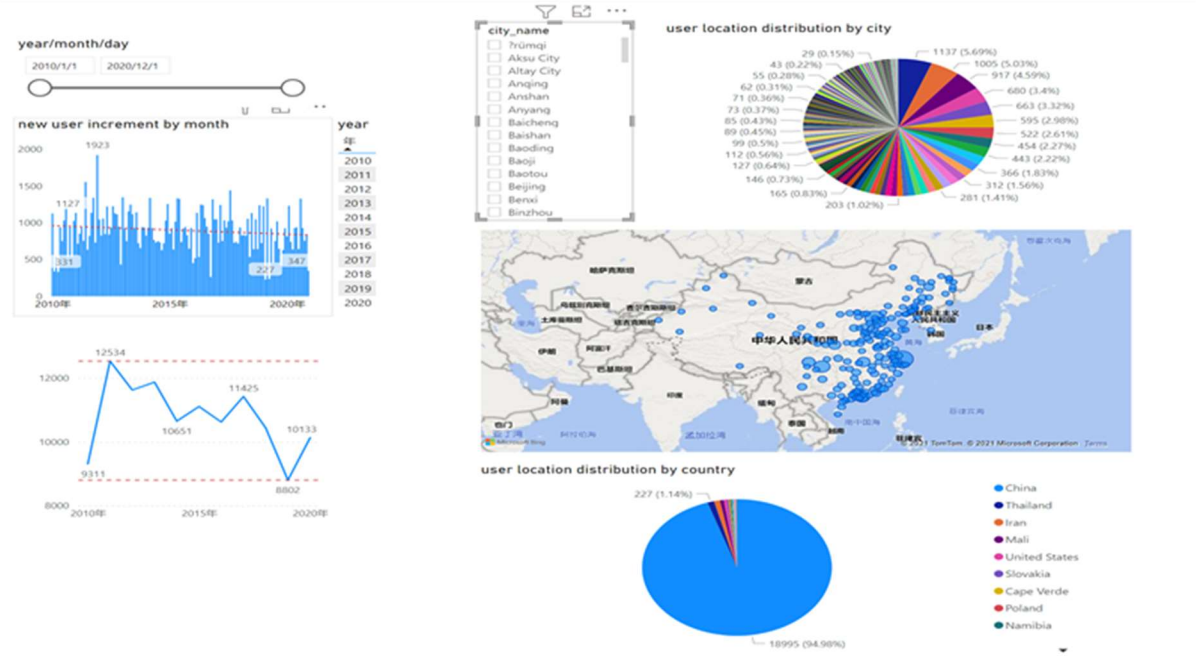
Fact Table				
Column Name	**Type**	**Key**	**Data Source (Table Name)**	**Data Path**
user_f				
date_key	int	Foreign Key (FK) – date_d	date_d, user	user.registration_date & date_d.y_m_d
location_key	int	FK – location_d	location_d, user, city	user.city_id & city.city_id + city.city_name & location_d.city_name
bcomment_f				
date_key	int	FK – date_d	date_d, book_user	book_user.rating_date & date_d.y_m_d
bcategory_key	int	FK – bcategory_d	bcategory_d, book_user	book_user.book_id & book_category.book_id + book_category.category_id & bcategory.bcategory_id + bcategory.name & bcategory_d.c1 and c2
rating	float		book_user	book_user.rating
mcomment_f				
date_key	int	FK – date_d	date_d, movie_user	movie_user.rating_date & date_d.y_m_d
mcategory_key	int	FK – bcategory_d	mcategory_d, movie_user	movie_user.movie_id & movie_category.movie_id + movie_category.category_id & mcategory.mcategory_id + mcategory.name & mcategory_d.c1 and c2
rating	float		movie_user	movie_user.rating

Dashboard

We use a cluster bar chart to show the growth of new users from 1 Jan 2010 to 31 Dec 2020, and a broken line chart to show the change in the number of new users each year. We saw the largest increase in new users in 2011 and the smallest increase in new users in 2019. Overall, the growth of new users is on a downward trend. The pie charts to show the proportion of new users in each city to

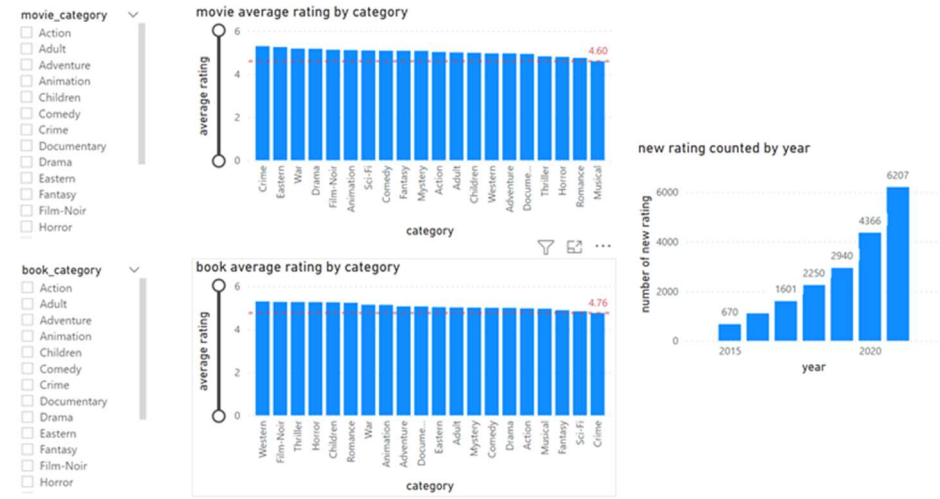
the total number of new users, and use map models to show the city of new users in China and the world.

The BI dashboard for user engagement is shown below. The dashboard has inbuilt filters that allow management to zoom in to specific time periods or cities.



In the user ratings section, we use a bar chart to show the data and rank the categories of books or movies in descending order of average ratings.

The BI dashboard for user ratings is shown below. The dashboard has inbuilt filters to allow management to select the movie and book categories they want to see.



Advertising revenue

【豆瓣网品牌类广告刊例价】

优质广告位* A level advertising spot

广告类型	广告标准	刊例价	购买单位	折扣/配送	广告位示意
图片广告 全站全流量*	310×188px 静态，gif/jpg≦ 16k（轮播）	¥30	CPM	有折扣，有配送	click
小组精准定向广告*	310×188px 静态，gif/jpg≦ 16k（轮播）	¥45	CPM	有折扣，有配送	click
地域定向 全站全流量*	310×188px 静态，gif/jpg≦ 16k（轮播）	¥39	CPM	有折扣，有配送	click

*注 1：全站全流量包含豆瓣社区、读书、电影、音乐等频道的一、二、三级页面统一尺寸广告位；

*注 2：小组精准定向可对 10 个以内中英文关键词进行精准投放，广告将会仅展示在关键词相关话题页面；

*注 3：地域精准定向可在全站全流量基础上对访问用户所在地域进行定向展示，目前仅对北/上/广三个地区定向；

Analytical questions

Regarding advertising revenue analytical questions:

- Firstly, we would be interested to find out if there are any discernible seasonal patterns for advertising revenue. For example, are there months that tend to generate higher ad bookings from advertisers?
- Secondly, which categories or sections tend to generate the most advertising revenue, besides Douban's homepage? (Homepage naturally commands the higher impressions given it is typically the point of landing or entry whereby visitors would tend to be on when they visit a website.)
- Thirdly, general trend patterns of consumers based on impressions garnered.
- Lastly, what is the type of data (impressions) distribution amongst the 4 ad types?

ETL

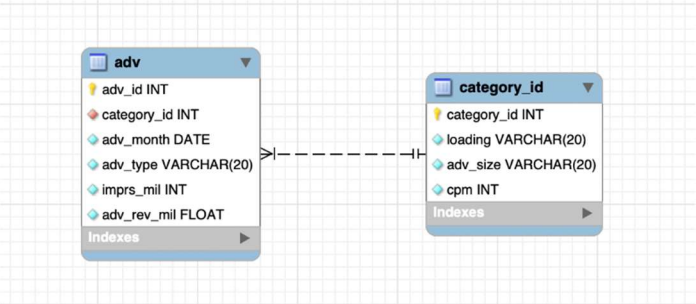
Code

```
# 1. Category_id table
CREATE TABLE `category_id` (
  `category_id` int NOT NULL AUTO_INCREMENT,
  `loading` varchar(20) NOT NULL,
  `adv_size` varchar(20) NOT NULL,
  `cpm` int NOT NULL,
```

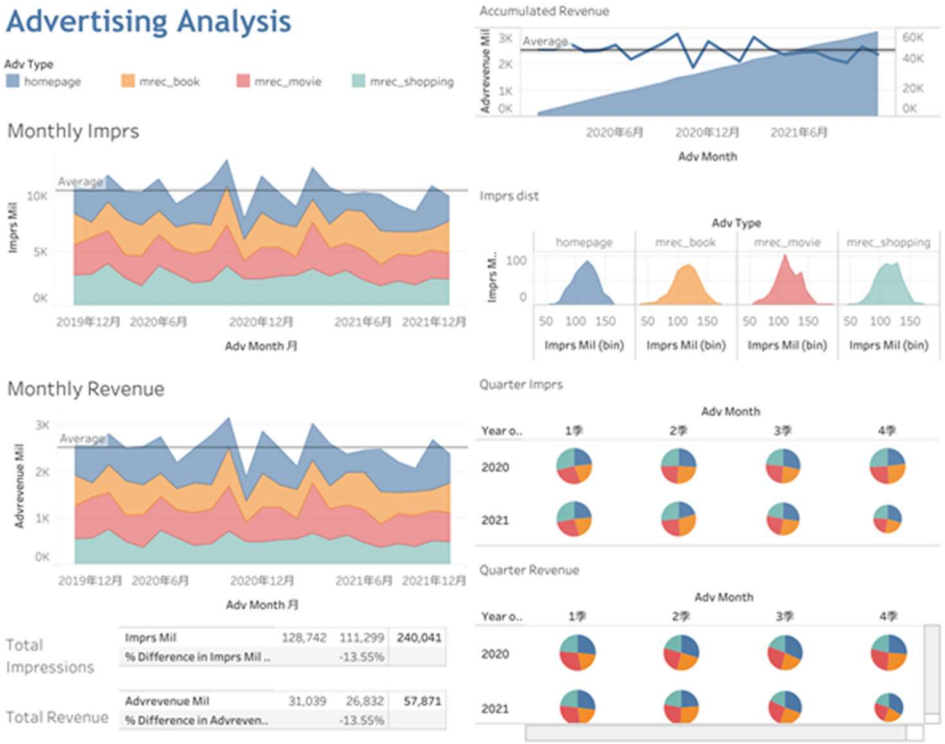
```
  PRIMARY KEY (`category_id`)
INSERT INTO `category_id` VALUES
(1,'before_login','950x90',275),(2,'ros_movie','300x250',260),(3,'ros_book','300x250',230),(4,'ros_shopping',
'300x250',200);

# 2. Advertise table
# import from csv file
CREATE TABLE `adv` (
  `adv_id` int NOT NULL AUTO_INCREMENT,
  `category_id` int NOT NULL,
  `adv_month` date NOT NULL,
  `adv_type` varchar(20) NOT NULL,
  `imprs_mil` int NOT NULL,
  `adv_rev_mil` float NOT NULL,
  PRIMARY KEY (`adv_id`)
```

OLAP data warehouse design



Dashboard



The dashboard reveals that advertising revenues tend to spike in June, October, and December, due to seasonal factors like national holidays. (Impressions of websites generally tend to be higher in these months, and Douban proves to be no exception.) Accumulated advertising revenue performance, though, has a slight downward sloping trend.

Interestingly enough, the impressions generated form a normal distribution $N(120,20)$ based on a total of 2,000 advertisements.

The decrease in revenue means that Douban should look into developing other revenue streams, for example its e-commerce revenue shown in the following section.

E-commerce revenue

Analytical questions

The important questions for e-commerce would be linked to the objective of generating more revenue.

- What is the general trend of our website's sales and which products are more popular?
- What is the profile of users buying our products?
- What books bring us the highest sales?
- How much is a customer willing to pay for a book?

ETL

Code	Comments
# 1. User dimension table CREATE TABLE olap.user_d (user_key INT NOT NULL AUTO_INCREMENT PRIMARY KEY) SELECT user_name, gender, city_id, DOB FROM oltp.user; ALTER TABLE olap.user_d ADD FOREIGN KEY (city_id) REFERENCES olap.location_d (location_key); ALTER TABLE olap.user_d CHANGE COLUMN city_id city_id INT NOT NULL	
# 2. Book dimension table #Create book dimension CREATE TABLE olap.book_d (book_key INT NOT NULL AUTO_INCREMENT PRIMARY KEY) SELECT book_name, rating, price FROM oltp.book; #Insert NA value for products, which do not have a book name ALTER TABLE olap.book_d AUTO_INCREMENT = 1; INSERT IGNORE book_d (book_key, book_name, rating, price) VALUES (NULL,"NA",0,0); ALTER TABLE olap.book_d CHANGE COLUMN price price DECIMAL(10,2) NOT NULL	In the OLTP, the items in the online cart could have a product_id or a book_id as the foreign key and there was no constraint set for the keys to be "NOT NULL", as such, in the book dimension table, an additional row (NA) was added to handle the NULL values that was extracted from the OLTP.
# 3. Product category and product dimension table #Create product category dimension	Similar to the book dimension table, an additional row was added to the product

CREATE TABLE olap.pcategory_d (
  pcategory_key INT NOT NULL AUTO_INCREMENT PRIMARY KEY
) SELECT category_name FROM
  oltp.product_category;
#Insert NA value for ebooks, which do not have a product category
ALTER TABLE olap.pcategory_d AUTO_INCREMENT = 1;
INSERT IGNORE olap.pcategory_d (pcategory_key, category_name)
VALUES (NULL,"NA");
#Create product dimension
CREATE TABLE olap.product_detail_d SELECT product_id AS product_key,
  product_name,
  pd.category_id,
  price FROM
  oltp.product_detail pd
  INNER JOIN
  oltp.product_category pc ON pc.category_id = pd.category_id;
#Insert NA value for ebooks, which do not have a product name
ALTER TABLE olap.product_detail_d AUTO_INCREMENT = 1;
INSERT IGNORE product_detail_d (product_key, product_name,
category_id, price)
VALUES (50,"NA",5,0);
ALTER TABLE olap.product_detail_d ADD PRIMARY KEY(product_key);
ALTER TABLE olap.product_detail_d
ADD FOREIGN KEY (category_id) REFERENCES olap.pcategory_d (pcategory_key);
	category and the product detail dimension tables to handle the NULL values in the OLTP. The tables were not combined for this reason.
	
	In addition, while we understand that the product_id should have been a surrogate key, the automatic sorting performed by mySQL when using the auto-increment would have resulted in a different total sales amount as the product_id used to extract the product price did not match the auto generated product_key, as such, we used the product_id in this case to act as the product_key.

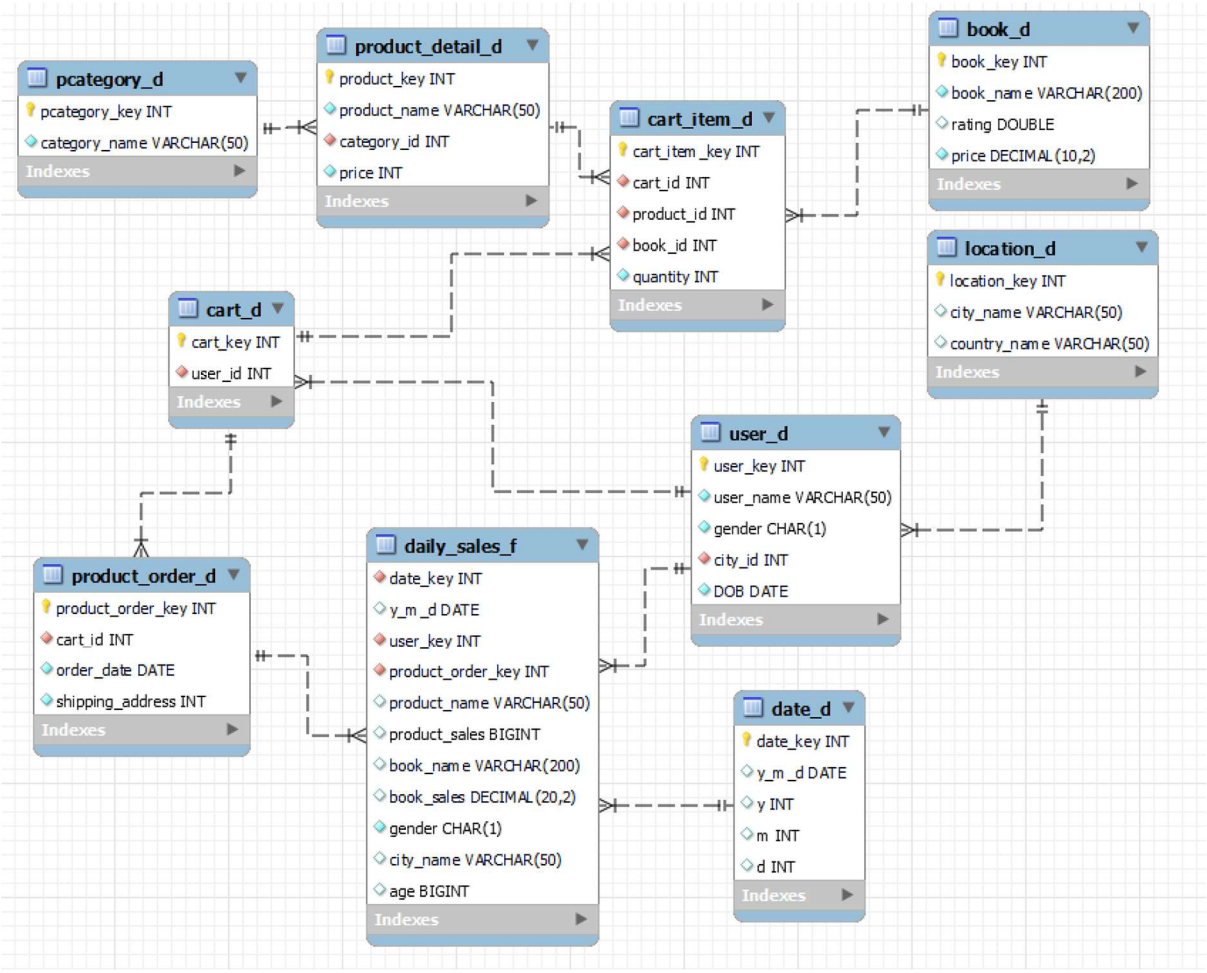
# 4. Cart and cart item dimension table 	
#Create cart dimension
CREATE TABLE olap.cart_d SELECT cart_id AS cart_key, user_id FROM
  oltp.product_cart;
ALTER TABLE olap.cart_d
ADD PRIMARY KEY (cart_key);
ALTER TABLE olap.cart_d
ADD FOREIGN KEY (user_id) REFERENCES olap.user_d (user_key);
#Create cart items dimension
CREATE TABLE olap.cart_item_d (
  cart_item_key INT NOT NULL AUTO_INCREMENT PRIMARY KEY
) SELECT c.cart_id AS cart_id, product_id, book_id, quantity FROM
  oltp.cart_item c;
#Replace NULL values in book/product category with the respective
NA category_id
SET SQL_SAFE_UPDATES = 0;
UPDATE olap.cart_item_d
SET
	We acknowledge that the cart_key should have been a surrogate key, however, the automatic sorting performed by SQL using the auto-increment would have resulted in a different user tagged to the cart, and we used the cart_id in this case to act as the cart_key.
	
	For the cart item dimension table, the products and books with NULL product_id or book_ID were replaced with the newly added NA values.

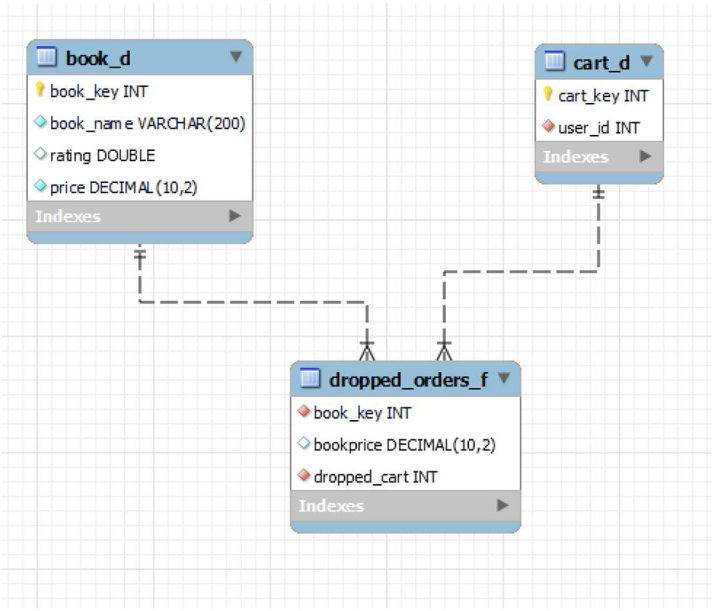
product_id = 50 WHERE product_id IS NULL; UPDATE olap.cart_item_d SET book_id = 501 WHERE book_id IS NULL; SET SQL_SAFE_UPDATES = 1; ALTER TABLE olap.cart_item_d ADD FOREIGN KEY (cart_id) REFERENCES olap.cart_d (cart_key); ALTER TABLE olap.cart_item_d ADD FOREIGN KEY (book_id) REFERENCES olap.book_d (book_key); ALTER TABLE olap.cart_item_d ADD FOREIGN KEY (product_id) REFERENCES olap.product_detail_d (product_key); ALTER TABLE olap.cart_item_d CHANGE COLUMN product_id product_id INT NOT NULL , CHANGE COLUMN book_id book_id INT NOT NULL;	
# 5. Order dimension table #Create order dimension CREATE TABLE olap.product_order_d (product_order_key INT NOT NULL AUTO_INCREMENT PRIMARY KEY) SELECT cart_id, order_date, shipping_address FROM oltp.product_order; ALTER TABLE olap.product_order_d ADD FOREIGN KEY (cart_id) REFERENCES olap.cart_d (cart_key)	
# 1. Daily sales fact table Create table olap.daily_sales_f AS SELECT d.date_key, d.y_m_d, u.user_key, po.product_order_key, pd.product_name, pd.price*ci.quantity as product_sales, b.book_name, b.price*ci.quantity as book_sales, u.gender, l.city_name, ceiling(DATEDIFF('2020-12-31',u.dob) / 365.25) as age FROM olap.product_order_d po INNER JOIN olap.cart_d pc on pc.cart_key = po.cart_id INNER JOIN olap.cart_item_d ci on ci.cart_id = pc.cart_key INNER JOIN olap.user_d u on pc.user_id = u.user_key INNER JOIN olap.location_d l on l.location_key = u.city_id INNER JOIN olap.date_d d on po.order_date = d.y_m_d LEFT JOIN olap.product_detail_d pd on pd.product_key = ci.product_id LEFT JOIN olap.book_d b on b.book_key = ci.book_id; ALTER TABLE olap.daily_sales_f ADD FOREIGN KEY (date_key) REFERENCES olap.date_d (date_key); ALTER TABLE olap.daily_sales_f ADD FOREIGN KEY (user_key) REFERENCES olap.user_d (user_key);	The grain that we used for our fact table was the individual item in our product orders.

ALTER TABLE olap.daily_sales_f ADD FOREIGN KEY (product_order_key) REFERENCES olap.product_order_d (product_order_key);	
# 2. Dropped Orders Fact Table Create table olap.dropped_orders_f as select b.book_key, b.price as bookprice, ci.cart_id as dropped_cart FROM olap.cart_d pc left join olap.product_order_d po on po.cart_id = pc.cart_key inner join olap.cart_item_d ci on pc.cart_key = ci.cart_id inner join olap.book_d b on b.book_key = ci.book_id where po.product_order_key is NULL and ci.product_id = 50 order by b.price; ALTER TABLE olap.dropped_orders_f ADD FOREIGN KEY (book_key) REFERENCES olap.book_d (book_key); ALTER TABLE olap.dropped_orders_f ADD FOREIGN KEY (dropped_cart) REFERENCES olap.cart_d (cart_key);	We also created a factless fact table to analyse the online carts that were abandoned and the books that were in them.

OLAP data warehouse design

The ER diagram for e-commerce is shown below.





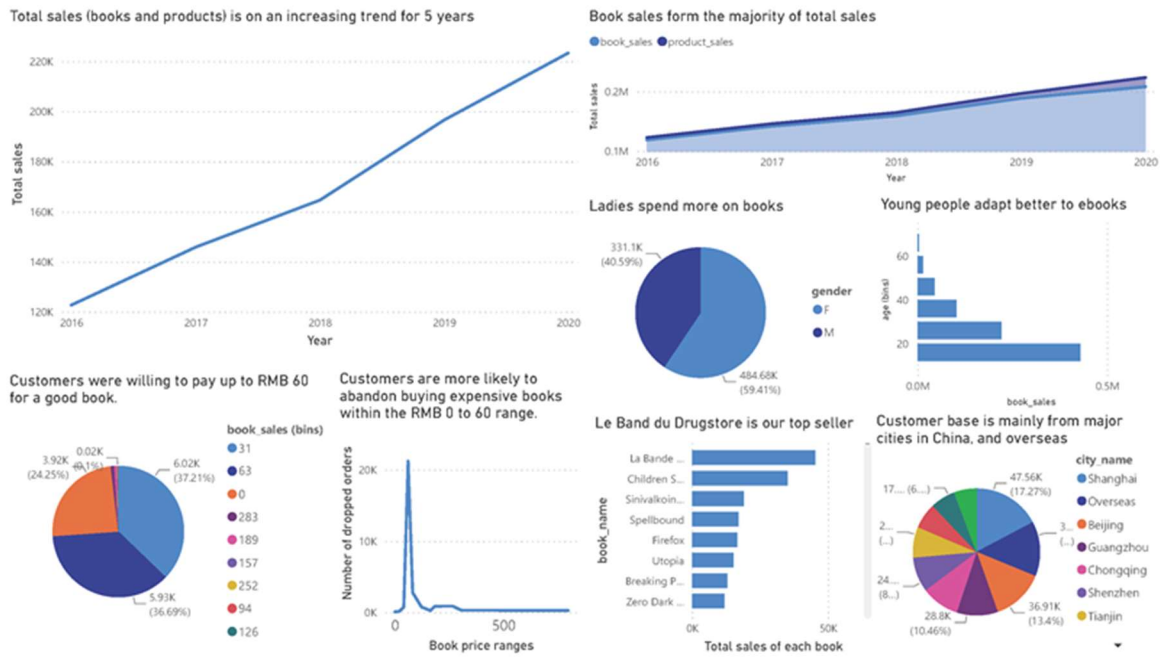
OLAP Table Structure

Dimension Table			
Column Name	**Data Type**	**Key**	**Data Source (Table Name)**
book_d			
book_key	int (auto-increment)	surrogate key	
book_name	varchar(200)		book
rating	double		book
price	decimal(10,2)		book
pcategory_d			
pcategory_key	int (auto-increment)	surrogate key	
category_name	varchar(50)		product_category
product_detail_d			
product_key	int	Primary Key (PK)	
product_name	varchar(50)		product_detail
category_id	int		product_detail
price	int		product_detail
cart_d			
cart_key	int	PK	
user_id	int		cart
cart_item_d			
cart_item_key	int (auto-increment)	surrogate key	
cart_id	int		cart_item
product_id	int		cart_item
book_id	int		cart_item
quantity	int		cart_item
user_d			
user_key	int (auto-increment)	surrogate key	
user_name	varchar(50)		user
gender	char(1)		user
city_id	int		user
DOB	date		user
product_order_d			
product_order_key	int (auto-increment)	surrogate key	
cart_id	int		product_order
order_date	date		product_order
shipping_address	int		product_order

Fact Table				
Column Name	**Data Type**	**Key**	**Data Source (Table Name)**	**Data Path**
daily_sales_f				
date_key	int	Foreign Key (FK) – date_d	date_d, product_order_d	product_order_d.order_date & date_d.y_m_d
y_m_d	date		product_order_d	product_order_d.order_date
user_key	int	FK – user_d	user_d	user_d.user_key
product_order_key	int	FK – product_order_d	product_order_d	product_order_d. product_order_key
product_name	varchar(50)		product_detail_d	product_order_d.cart_id & cart_d.cart_key + cart_d.cart_key & cart_item_d.cart_id + cart_item_d.product_id & product_detail_d.product_key
product_sales	bigint		product_detail_d & cart_item_d	product_order_d.cart_id & cart_d.cart_key + cart_d.cart_key & cart_item_d.cart_id + cart_item_d.product_id & product_detail_d.product_key
book_name	varchar(200)		book_d	product_order_d.cart_id & cart_d.cart_key + cart_d.cart_key & cart_item_d.cart_id + cart_item_d.book_id & book_d.book_key
book_sales	decimal(20,2)		book_d & cart_item_d	product_order_d.cart_id & cart_d.cart_key + cart_d.cart_key & cart_item_d.cart_id + cart_item_d.book_id & book_d.book_key
gender	char(1)		user_d	product_order_d.cart_id & cart_d.cart_key + cart_d.user_id & user_d.user_key
city_name	varchar(50)		user_d & location_d	product_order_d.cart_id & cart_d.cart_key + cart_d.user_id & user_d.user_key + user_d.city_id & location_d.location_key
age	bigint		user_d	product_order_d.cart_id & cart_d.cart_key + cart_d.user_id & user_d.user_key
dropped_orders_f				
book_key	int	FK – book_d	book_d	product_order_d.cart_id & cart_d.cart_key + cart_d.cart_key & cart_item_d.cart_id id + cart_item_d.book_id & book_d.book_key
bookprice	decimal(20,2)		book_d	product_order_d.cart_id & cart_d.cart_key + cart_d.cart_key & cart_item_d.cart_id id + cart_item_d.book_id & book_d.book_key
dropped_cart	int	FK – cart_d	cart_id & product_order_d	product_order_d.cart_id & cart_d.cart_key

Dashboard

The dashboard below answers the pertinent questions raised about Douban's e-commerce revenue.



From the dashboard, we can see the sales as a whole have been rising for the past 5 years.[6] Most of it comes from ebooks instead of merchandise. The typical customer of our ebooks is a lady around her 20s, living in Shanghai and Beijing in China, or even overseas. She is willing to buy good books for up to RMB 60, because free books are not actually our most popular product. However, within the RMB 0 to 60 range, our customers are still price sensitive and tend to abandon buying books which are closer to the maximum of the range, around RMB 60.[7]

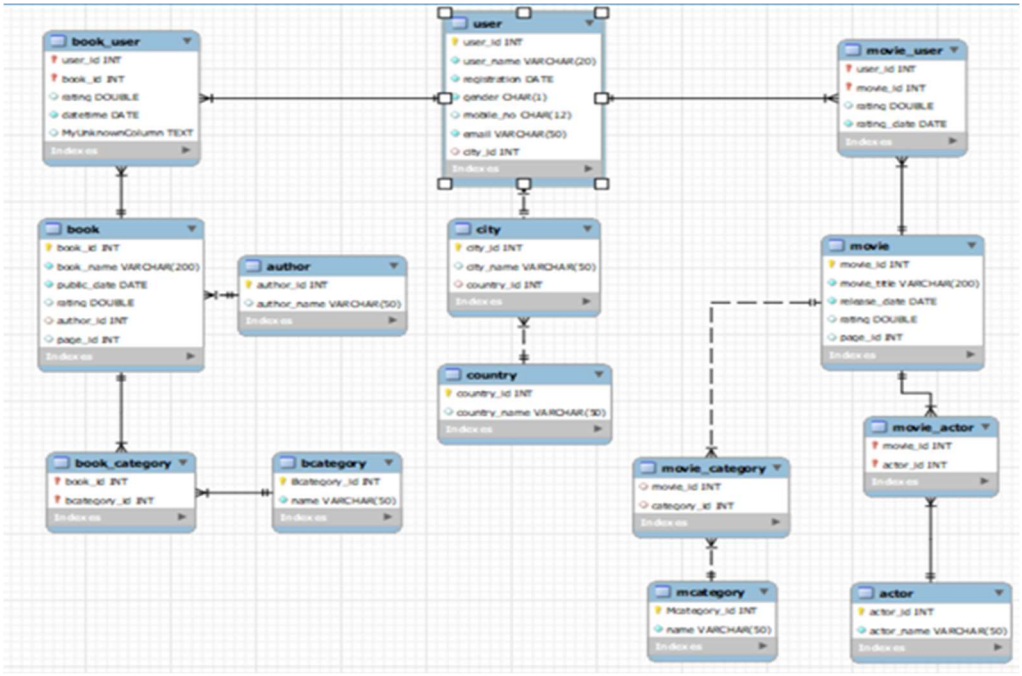
Despite the price sensitivity, we can see that the book that brought us the most sales Le Band du Drugstore. Based on how we generated the book prices, the price point of the book happens to be a book that was priced highly on Amazon at USD 63 i.e., Harry Potter: The Complete Collection (1-7). Therefore, there is still value to launch collections which are priced higher, as long as they are popular enough.

[6] https://www.spglobal.com/marketintelligence/en/news-insights/blog/amazon-ecommerce-sales-soar-amid-covid19

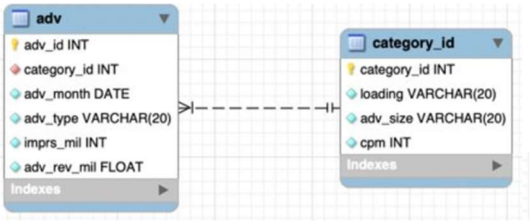
[7] https://www.optimizely.com/optimization-glossary/shopping-cart-abandonment/

Annex

OLTP diagram – Review platform for books and movies



OLTP diagram – Advertising revenue



OLTP diagram – E-commerce revenue