

DATA MINING

**REPORT**

*Data Preprocessing, Cleaning, and Database Schema*

*Design for Amazon Fashion Products*

**Data Preprocessing, Cleaning:**

After converting the json file into a dataframe to do preprocessing, there were multiple steps to clean the data accordingly. Below is a brief breakdown of the steps involved:

* Changing column names
* Cleaning numeric data
* Changing their data type
* Removing $ from price and unnecessary text from columns
* Drop duplicates and remove nan values.
* Make columns for ids to be used in the database.
* On columns: description and reviews, many steps were involved. For example: lowercase, stopwords, special characters, stemming, lemmatizing and removing emojis+urls from text.
* My review column consists of reviewer name , date and review text. I made 3 separate columns for that and cleaned both review\_text and reviewer name by the same steps mentioned above.

**Database Schema Design for Amazon Fashion Products:**

Columns i had in my dataframe were:

* ['url', 'ASIN', 'gender', 'manufacturer', 'rating', 'category', 'subCategory', 'price', 'title', 'brandName', 'images', 'color\_imgs', 'colors', 'review\_date', 'decs', 'clean\_review', 'r\_names' , ‘review\_id’ ,’category\_id’ , ‘image\_id’,’img\_path’].
* On the bases of above columns , i made the following schema for database:
* CREATE TABLE cc(

category\_id VARCHAR(40) PRIMARY KEY,

category VARCHAR(255) NOT NULL,

subcategory VARCHAR(60) NOT NULL

);

* CREATE TABLE iii(

image\_id VARCHAR(40) PRIMARY KEY,

imageurl VARCHAR(1255) NOT NULL,

clr\_images VARCHAR(4250) NOT NULL,

Imgpath VARCHAR(300) NOT NULL

);

* CREATE TABLE r(

review\_id VARCHAR(40) PRIMARY KEY ,

reviewer\_name VARCHAR(500) NOT NULL,

review\_date VARCHAR(500) NOT NULL,

review\_text TEXT NOT NULL

);

* CREATE TABLE p(

asin VARCHAR(20) PRIMARY KEY,

title VARCHAR(255) NOT NULL,

descriptions TEXT NOT NULL,

rating FLOAT NOT NULL,

color VARCHAR(1250) NOT NULL,

brand\_name VARCHAR(255) NOT NULL,

price DECIMAL(10, 2) NOT NULL,

manufacturer VARCHAR(255) NOT NULL,

gender VARCHAR(10) NOT NULL,

url VARCHAR(1255) NOT NULL,

category\_id VARCHAR(40),

image\_id VARCHAR(40),

review\_id VARCHAR(40),

FOREIGN KEY (review\_id) REFERENCES r(review\_id),

FOREIGN KEY (category\_id) REFERENCES category(category\_id),

FOREIGN KEY (image\_id) REFERENCES images(image\_id)

);

**QUERIES:**

Few of the queries i performed on the above tables were:

SELECT p.\*, cc.category, iii.imageurl

FROM p

LEFT JOIN cc ON p.category\_id = cc.category\_id

LEFT JOIN iii ON p.image\_id = iii.image\_id;

SELECT p.asin, p.title, AVG(p.rating) AS avg\_rating

FROM p

LEFT JOIN r ON p.asin = r.review\_id

GROUP BY p.asin, p.title;

SELECT p.asin, p.title, p.color

FROM p

WHERE p.color = 'red';

SELECT r.reviewer\_name, r.review\_date, r.review\_text

FROM r

ORDER BY r.review\_date DESC

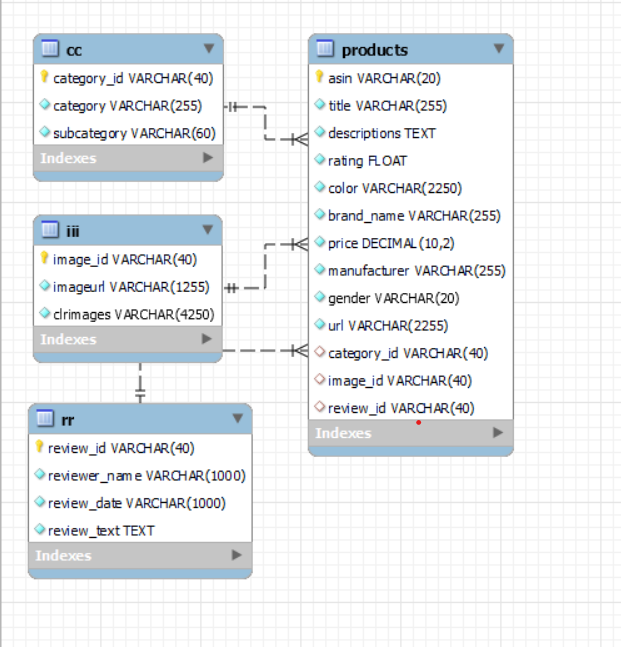
LIMIT 5;

P = products table

R = reviews table

Ii = image table

Cc = category table



# 